

# **Aula 00 - Prof. Diego Carvalho e Fernando Pedrosa**

*SMAP Porto Alegre-RS (Analista de  
Tecnologia da Informação e  
Comunicação) Engenharia de Software -  
Autor:  
2024 (Pós-Edital)*  
**Diego Carvalho, Emannelle  
Gouveia Rolim, Equipe Informática  
2 (Diego Carvalho), Fernando**

**Pedrosa Lopes**  
27 de Outubro de 2024

# Índice

1) Apresentação do Prof. Diego Carvalho - Informática .....	3
2) Apresentação Flashcards .....	5
3) Engenharia de Requisitos .....	7
4) Engenharia de Requisitos - Classificação de Requisitos .....	11
5) Engenharia de Requisitos - Engenharia de Requisitos .....	24
6) Resumo - Engenharia de Requisitos .....	51
7) Questões Comentadas - Engenharia de Requisitos - CESPE .....	57
8) Questões Comentadas - Engenharia de Requisitos - FCC .....	108
9) Questões Comentadas - Engenharia de Requisitos - FGV .....	135
10) Questões Comentadas - Engenharia de Requisitos - Multibancas .....	151
11) Lista de Questões - Engenharia de Requisitos - CESPE .....	203
12) Lista de Questões - Engenharia de Requisitos - FCC .....	222
13) Lista de Questões - Engenharia de Requisitos - FGV .....	239
14) Lista de Questões - Engenharia de Requisitos - Multibancas .....	248
15) MVP .....	273
16) MVP - Questões Comentadas .....	277
17) MVP - Lista de Questões .....	281
18) Débito Técnico .....	284
19) Técnicas de Priorização de Backlog - Teoria .....	289
20) Técnicas de Priorização de Backlog - Questões Comentadas .....	303
21) Técnicas de Priorização de Backlog - Lista de Questões .....	313



## APRESENTAÇÃO DO PROFESSOR

# PROF. DIEGO CARVALHO

FORMADO EM CIÊNCIA DA COMPUTAÇÃO PELA UNIVERSIDADE DE BRASÍLIA (UNB), PÓS-GRADUADO EM GESTÃO DE TECNOLOGIA DA INFORMAÇÃO NA ADMINISTRAÇÃO PÚBLICA E, ATUALMENTE, AUDITOR FEDERAL DE FINANÇAS E CONTROLE DA SECRETARIA DO TESOURO NACIONAL.

## ESTRATÉGIA CONCURSOS

 PROFESSOR DIEGO CARVALHO - [WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO](https://www.instagram.com/professordiegovalho)



**Sobre o curso: galera, todos os tópicos da aula possuem Faixas de Incidência, que indicam se o assunto cai muito ou pouco em prova.** Diego, se cai pouco para que colocar em aula? Cair pouco não significa que não cairá justamente na sua prova! A ideia aqui é: se você está com pouco tempo e precisa ver somente aquilo que cai mais, você pode filtrar pelas incidências média, alta e altíssima; se você tem tempo sobrando e quer ver tudo, vejam também as incidências baixas e baixíssimas. *Fechado?*

INCIDÊNCIA EM PROVA: BAIXÍSSIMA

INCIDÊNCIA EM PROVA: BAIXA

INCIDÊNCIA EM PROVA: MÉDIA

INCIDÊNCIA EM PROVA: ALTA

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Além disso, essas faixas não são por banca – é baseado tanto na quantidade de vezes que caiu em prova independentemente da banca quanto nas minhas próprias avaliações sobre cada assunto.



#ATENÇÃO

# Avisos Importantes



## O curso abrange todos os níveis de conhecimento...



Esse curso foi desenvolvido para ser acessível a **alunos com diversos níveis de conhecimento diferentes**. Temos alunos mais avançados que têm conhecimento prévio ou têm facilidade com o assunto. Por outro lado, temos alunos iniciantes, que nunca tiveram contato com a matéria ou até mesmo que têm trauma dessa disciplina. A ideia aqui é tentar atingir ambos os públicos - iniciantes e avançados - da melhor maneira possível..

## Por que estou enfatizando isso?



O **material completo** é composto de muitas histórias pessoais, exemplos, metáforas, piadas, memes, questões, desafios, esquemas, diagramas, imagens, entre outros. Já o **material simplificado** possui exatamente o mesmo núcleo do material completo, mas ele é menor e mais objetivo. *Professor, eu devo estudar por qual material?* Se você quiser se aprofundar nos assuntos ou tem dificuldade com a matéria, necessitando de um material mais passo-a-passo, utilize o material completo. Se você não quer se aprofundar nos assuntos ou tem facilidade com a matéria, necessitando de um material mais direto ao ponto, utilize o material simplificado.



## Por fim...

**O curso contém diversas questões espalhadas em meio à teoria.** Essas questões possuem um comentário mais simplificado porque **têm o único objetivo de apresentar ao aluno como bancas de concurso cobram o assunto previamente administrado**. A imensa maioria das questões para que o aluno avalie seus conhecimentos sobre a matéria estão dispostas ao final da aula na lista de exercícios e **possuem comentários bem mais abrangentes**.





# ESTRATÉGIA FLASHCARDS

📱 Você tem dificuldade de estudar, memorizar e revisar os conteúdos que estuda em nossas aulas? Então nós temos a ferramenta perfeita para você!

Apresentamos o **Estratégia Cards**: app de flashcards que vai revolucionar sua forma de **estudar** e **revisar** conteúdos de provas de concurso público. Com nossa tecnologia inovadora e interface amigável, você dominará os tópicos mais complexos de maneira eficiente e divertida.

## 🌟 Recursos do Estratégia Cards:

<b>Curadoria de Flashcards</b>	Flashcards criados e revisados por professores especializados em cada área, com qualidade e voltados para concursos públicos.
<b>Flashcards Personalizados</b>	Crie seus próprios flashcards, cobrindo os principais tópicos e matérias dos concursos públicos.
<b>Repetição Espaçada</b>	Técnica de aprendizagem que envolve revisar informações em intervalos crescentes para melhorar a retenção de longo prazo e combater o esquecimento.
<b>Estatísticas Personalizadas</b>	Visualize graficamente o percentual de acertos, erros ou dúvidas dos decks estudados.
<b>Modo Offline</b>	Estude em qualquer lugar, mesmo sem conexão à internet, fazendo o download dos decks.
<b>Estudo por Áudio</b>	<i>Está dirigindo ou fazendo esteira e quer continuar estudando?</i> Basta utilizar a opção “Escutar”.
<b>Decks Favoritos</b>	Você pode escolher decks específicos como favoritos e visualizá-los em uma aba separada do app.
<b>Opções de Estudo</b>	Você poderá estudar todos os cards de um deck; ou apenas os que você errou; ou apenas os que você não estudou ainda; entre outras opções.

## 📱 E como eu consigo baixar?



É muito fácil! Basta pesquisar por “Estratégia Cards” na loja oficial do seu smartphone.

Se você tiver um Android, basta acessar a **Google Play**;



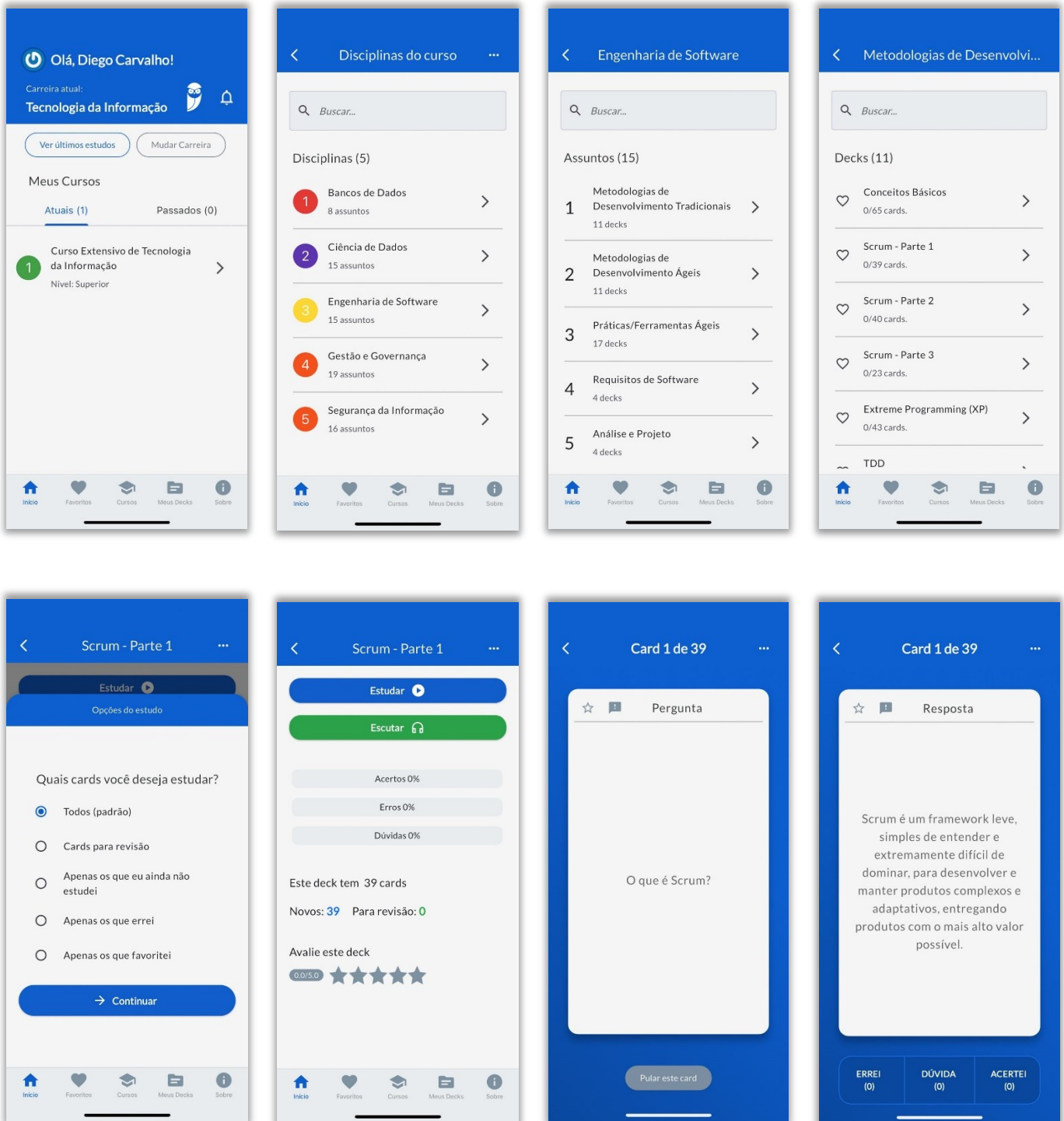
Se for tiver um iPhone, basta acessar a **App Store (iOS)**.



## É para acessar?

Para acessar, basta ter uma conta no Estratégia Concursos. Em seguida, utilize suas credenciais de login e senha para acessar o aplicativo. Por fim, acessa a carreira de Tecnologia da Informação.

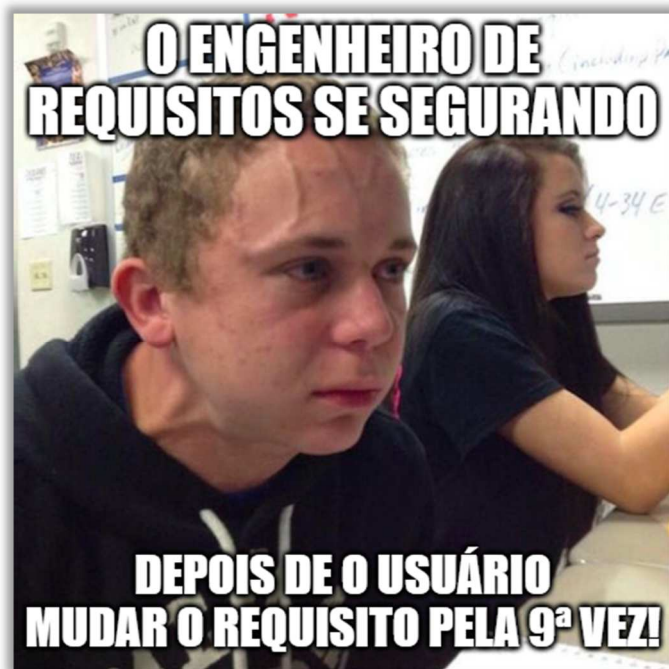
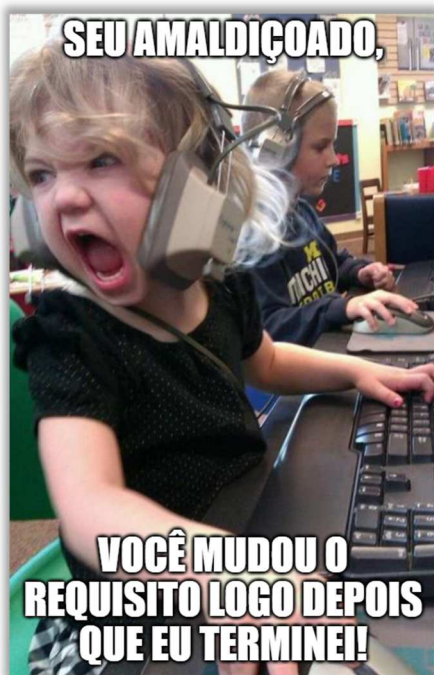
## Como utilizar o app:



## APRESENTAÇÃO

Seus maravilhosos, essa aula é **essencial** para todos os concursos! Eu não consigo descrever a importância desse conteúdo – é absolutamente importantíssimo que vocês dominem esse assunto. Se esse conteúdo está no edital, há 99,999999999999% de chances de ele cair na prova. *Por que? Porque não é possível pensar em software algum sem pensar nos requisitos.* Bacana? Então vamos lá, não quero ver ninguém desanimado :)

 **PROFESSOR DIEGO CARVALHO - [WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO](http://WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO)**



**Galera, todos os tópicos da aula possuem Faixas de Incidência, que indicam se o assunto cai muito ou pouco em prova.** *Diego, se cai pouco para que colocar em aula?* Cair pouco não significa que não cairá justamente na sua prova! A ideia aqui é: se você está com pouco tempo e precisa ver somente aquilo que cai mais, você pode filtrar pelas incidências média, alta e altíssima; se você tem tempo sobrando e quer ver tudo, vejam também as incidências baixas e baixíssimas. *Fechado?*

INCIDÊNCIA EM PROVA: BAIXÍSSIMA

INCIDÊNCIA EM PROVA: BAIXA

INCIDÊNCIA EM PROVA: MÉDIA

INCIDÊNCIA EM PROVA: ALTA

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Além disso, essas faixas não são por banca – é baseado tanto na quantidade de vezes que caiu em prova independentemente da banca e também em minhas avaliações sobre cada assunto...





#ATENÇÃO

# Avisos Importantes



## O curso abrange todos os níveis de conhecimento...

Esse curso foi desenvolvido para ser acessível a **alunos com diversos níveis de conhecimento diferentes**. Temos alunos mais avançados que têm conhecimento prévio ou têm facilidade com o assunto. Por outro lado, temos alunos iniciantes, que nunca tiveram contato com a matéria ou até mesmo que têm trauma dessa disciplina. A ideia aqui é tentar atingir ambos os públicos - iniciantes e avançados - da melhor maneira possível..



## Por que estou enfatizando isso?

O **material completo** é composto de muitas histórias, exemplos, metáforas, piadas, memes, questões, desafios, esquemas, diagramas, imagens, entre outros. Já o **material simplificado** possui exatamente o mesmo núcleo do material completo, mas ele é menor e bem mais objetivo. *Professor, eu devo estudar por qual material?* Se você quiser se aprofundar nos assuntos ou tem dificuldade com a matéria, necessitando de um material mais passo-a-passo, utilize o material completo. Se você não quer se aprofundar nos assuntos ou tem facilidade com a matéria, necessitando de um material mais direto ao ponto, utilize o material simplificado.



## Por fim...

O curso contém diversas questões espalhadas em meio à teoria. Essas questões possuem um comentário mais simplificado porque **têm o único objetivo de apresentar ao aluno como bancas de concurso cobram o assunto previamente administrado**. A imensa maioria das questões para que o aluno avalie seus conhecimentos sobre a matéria estão dispostas ao final da aula na lista de exercícios e **possuem comentários bem mais completos, abrangentes e direcionados**.

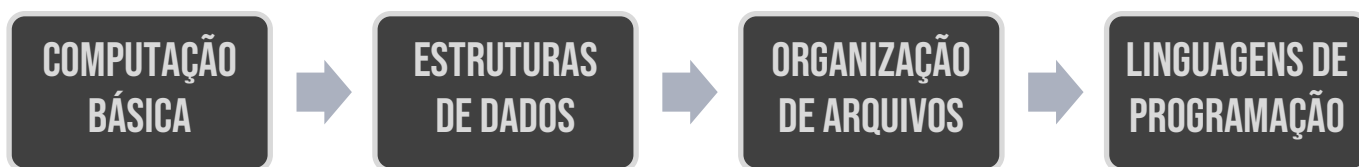


## REQUISITOS

### Conceitos Básicos

INCIDÊNCIA EM PROVA: BAIXA

Pessoal, vamos começar beeeeeeeem o básico do básico! *O que significa o verbo "requisitar"?* **De acordo com o dicionário, requisitar é pedir ou requerer algo ou alguém de modo oficial ou formal.** É solicitar alguma coisa, geralmente de forma temporária; é exigir. Já o substantivo "requisito" é aquilo que é necessário e indispensável; uma condição indispensável; uma exigência. Beleza, isso é o básico – agora nós vamos ver alguns exemplos.



**Vamos imaginar um exemplo da saudosa época de faculdade! Início de semestre, é a hora de fazer a matrícula.** Vejam o fluxo de disciplinas apresentado acima do curso de Ciência da Computação da Universidade de Brasília (UnB), onde eu fiz a minha graduação! Galera, no currículo do curso, ter feito a disciplina de Computação Básica é requisito para fazer a disciplina de Estrutura de Dados.

Assim como ter feito a disciplina de Estrutura de Dados é requisito para fazer a disciplina de Organização de Arquivos; e ter feito a Organização de Arquivos é requisito para fazer Linguagens de Programação. Se você estiver no primeiro semestre, você não pode fazer a disciplina de Organização de Arquivos, porque você obrigatoriamente precisa ter feito Computação Básica e Estrutura de Dados. **Notem que uma matéria é exigência para outra.**

SER BRASILEIRO NATO

TER IDADE SUPERIOR A 35 ANOS

PLENO EXERCÍCIO DOS DIREITOS POLÍTICOS

SER ELEITOR E TER DOMICÍLIO ELEITORAL NO BRASIL

SER FILIADO A UMA AGREMIÇÃO OU PARTIDO POLÍTICO

NÃO TER SUBSTITUÍDO O ATUAL PRESIDENTE NOS SEIS MESES ANTES DA ELEIÇÃO

Na verdade, no momento da matrícula de uma determinada matéria, é possível verificar quais são os pré-requisitos exigidos para fazê-la. *Querem outro exemplo?* **Vamos para a área jurídica! Algum**



**de vocês quer mudar o Brasil e decide se candidatar à Presidente da República!** No entanto, não é simples assim... é necessário cumprir uma série de requisitos para poder sequer concorrer a esse cargo – como mostra a imagem a anterior.

Para ser candidato você deve ser brasileiro nato; ter idade superior a trinta a cinco anos; estar em pleno exercício dos direitos políticos; ser eleitor e ter domicílio eleitoral no Brasil; ser filiado a uma agremiação ou partido político; e não ter substituído o atual presidente nos seis meses antes da eleição. **Satisfeitos esses requisitos, você – revolucionário que irá mudar nossa nação – poderá ser candidato à presidência.** *Bacana, pessoal?*

*Viram como é tranquilo de compreender o que é um requisito? É um conceito comum que faz parte do nosso cotidiano!* No contexto de Engenharia, costuma-se defini-lo como uma propriedade ou um comportamento que um produto ou serviço deve atender. Já no contexto de Engenharia de Software, costuma-se defini-lo como uma condição que deve ser satisfeita para se alcançar um objetivo, ou a qualidade do sistema que deve ser provida para ser útil a seus usuários.





## Classificação de Requisitos

Galera, existem diversas maneiras diferentes de se classificar requisitos de software! Veremos a seguir algumas dessas classificações:

### Classificação quanto ao nível de abstração

INCIDÊNCIA EM PROVA: BAIXA

Vamos iniciar falando sobre a classificação quanto ao nível de abstração. *O que é abstração? Em resumo, é a subtração de detalhes. Logo, se algo é muito abstrato, então é pouco detalhado; se algo é pouco abstrato, então é muito detalhado. Bacana?* Essa classificação se divide em Requisitos de Usuário e Requisitos de Sistema – evidentemente que o primeiro é mais abstrato que o segundo. Vejam só...

- **Requisitos de Usuário:** descrições, em linguagem natural e com diagramas, de quais serviços o sistema deve fornecer e as restrições sob as quais deve operar. São requisitos com alto nível de abstração e poucos detalhes, feitos para serem lidos por pessoas leigas – podem ser funcionais ou não funcionais (veremos essa classificação mais à frente).

**Exemplo:** o sistema deve gerar um relatório de acompanhamento mensal e enviar para os usuários por e-mail – notem que há poucos detalhes e nada muito técnico.

- **Requisitos de Sistema:** descrições detalhadas sobre as funções, operações e restrições de sistema que definem exatamente o que deve ser implementado. São requisitos com baixo nível de abstração e muitos detalhes, feitos para serem lidos por pessoas experientes – podem ser funcionais ou não funcionais (veremos essa classificação mais à frente).

**Exemplo:** o sistema deve gerar um relatório com índices a partir de *views* materializadas gerados a partir de um banco multidimensional – bem mais detalhes e bem mais técnico.

*Professor, isso é só para complicar a minha vida?* Não, pessoal – por incrível que pareça, isso tem a sua utilidade! **Ao escrever requisitos, deve-se considerar quem serão seus leitores, uma vez que eles possuem níveis diferentes de conhecimento, portanto os requisitos devem ter níveis diferentes de detalhamento.** Usuários geralmente não dão a mínima para como o sistema será implementado; já programadores necessitam de detalhes mais técnicos e precisos.

**(TCE/PR – 2011)** No processo de engenharia de requisitos, os tipos de requisitos de usuário e de sistema podem ser, respectivamente,

- a) apenas funcionais; apenas não funcionais.
- b) apenas não funcionais; apenas funcionais.
- c) apenas funcionais; funcionais e não funcionais.
- d) funcionais e não funcionais; apenas não funcionais.



e) funcionais e não funcionais; funcionais e não funcionais.

---

**Comentários:** ambos podem ser tanto funcionais quanto não-funcionais (Letra E).

**(MEC – 2015)** Os requisitos de usuários são mais específicos que os requisitos de sistemas, uma vez que estes últimos são utilizados para expressar o que o sistema deve implementar.

---

**Comentários:** na verdade, os requisitos de sistema são mais específicos e detalhados que os requisitos de usuário – o primeiro são descrições mais detalhadas das funções, serviços e restrições operacionais do sistema de software; e o segundo são declarações, em uma linguagem natural com diagramas, de quais serviços o sistema deverá fornecer a seus usuários e as restrições com as quais este deve operar (Errado).

**(SERPRO – 2013)** Um exemplo de requisito de sistema bem descrito é: “O sistema deve fornecer um gráfico comparativo entre as vendas previstas e as realizadas”.

---

**Comentários:** esse é um exemplo de requisitos de usuário e, não, de sistema (Errado).

**(MEC – 2011)** A documentação de requisitos deve conter duas perspectivas: uma voltada para o cliente, em linguagem compreensível por ele, e outra voltada para o desenvolvedor, em uma linguagem técnica de modelagem.

---

**Comentários:** a questão trata respectivamente dos requisitos de usuário e dos requisitos de sistema (Correto).

**(IPHAN – 2018) Situação hipotética:** Ao efetuar a especificação de requisitos, um analista abrangeu os requisitos de usuário e os de sistema, tendo incluído entre os requisitos de usuário os funcionais e os não funcionais. **Assertiva:** A especificação realizada pelo analista está correta, pois os requisitos não funcionais podem ser inseridos nos requisitos de usuário.

---

**Comentários:** requisitos de usuário e de sistema podem ser funcionais ou não-funcionais (Correto).



## Classificação quanto à qualidade

INCIDÊNCIA EM PROVA: BAIXA

Vamos falar sobre a classificação quanto à qualidade, mas antes nós vamos ver um conceito importante: *Quality Function Deployment* (QFD). Traduzido como Disponibilização da Função de Qualidade, **trata-se de uma técnica de gestão da qualidade aplicada ao levantamento de requisitos que traduz as necessidades do cliente em requisitos técnicos** buscando maximizar a satisfação do cliente e enfatizando o entendimento do que é valioso para o cliente.

- **Requisitos Normais:** refletem os objetivos e metas estabelecidos para um produto ou sistema durante reuniões com o cliente. Se esses requisitos estiverem presentes, o cliente fica satisfeito. Exemplos de Requisitos Normais poderiam ser tipos de displays gráficos solicitados, funções de sistema específicas e níveis de desempenho definidos.

**Requisitos normais são exatamente o que a palavra diz: são aqueles requisitos normais, comuns, corriqueiros, etc.** Imaginem que vocês estão levantando requisitos para um sistema de livros de uma biblioteca. Considerando esse contexto, um requisito normal poderia ser por exemplo: “*cadastrar livros recebidos*” ou “*permitir buscar um livro pelo seu título*”. Enfim... são requisitos completamente normais para o contexto do negócio.

- **Requisitos Esperados:** estão implícitos no produto ou sistema e podem ser tão fundamentais que o cliente não os declara explicitamente. Sua ausência será causa de grande insatisfação. Exemplos de Requisitos Esperados: facilidade na interação homem-máquina, confiabilidade e correção operacional global e facilidade na instalação do software.

Nós vimos que um requisito normal seria permitir a busca por um livro a partir do seu título. Agora imaginem que essa busca retorne que encontrou dezenas de livros com o título que eu inseri, mas não informa quais – apenas diz que encontrou. **Galera, é evidente que eu não quero saber apenas que livros foram encontrados a partir de um título, eu também quero saber quais livros foram encontrados.** *Então, o que seria um requisito esperado?*

**Um requisito esperado seria um relatório com todos os livros encontrados a partir do título de entrada.** Se o sistema simplesmente diz que encontrou livros com o título inserido, mas não apresenta nenhum relatório com os livros encontrados, isso pode gerar uma grande insatisfação no cliente, porque esse é um requisito esperado que ele sequer precisa especificar para os desenvolvedores. *Entenderam agora?*

- **Requisitos Fascinantes:** esses recursos vão além da expectativa dos clientes e demonstram ser muito satisfatórios quando presentes. Por exemplo, o software para um novo celular vem com recursos-padrão, mas junto vem um conjunto de capacidades não esperadas. Exemplos de Requisitos Fascinantes: tecla multitoque e correio de voz visual.

Imaginem que o programador implementou o cadastro dos livros, implementou um filtro para que você possa buscá-los com diversos parâmetros diferentes, implementou também o relatório dos



livros encontrados, mas – pensando que se trata de uma biblioteca – **ele implementou tudo isso funcionando com tecnologia assistiva que pode ajudar portadores de necessidades especiais, por exemplo com um leitor de tela para cegos, por exemplo.**

Notem que, por vezes, nem o usuário tinha pensado em pedir isso, mas o programador foi lá e implementou – isso é um requisito fascinante, porque ele supera as expectativas dos clientes.

**(SINESP – 2015)** A disponibilização da função de qualidade (Quality Function Deployment, QFD) é uma técnica aplicável à atividade de levantamento de requisitos a qual traduz as necessidades do cliente para requisitos técnicos de software. Esta técnica classifica as necessidades em requisitos:

- a) essenciais, reais e complexos.
- b) reais, complexos e normais.
- c) complexos, fascinantes e essenciais.
- d) fascinantes, esperados e reais.
- e) normais, esperados e fascinantes.

**Comentários:** trata-se dos requisitos normais, esperados e fascinantes (Letra E).



## Classificação quanto à evolução

INCIDÊNCIA EM PROVA: BAIXA

**Agora nós vamos falar sobre a classificação de requisitos quanto à evolução.** Galera, essa classificação divide os requisitos em requisitos permanentes e requisitos voláteis. Vejamos...

- **Requisitos Permanentes:** também chamados de Requisitos Estáveis, estão diretamente ligados a atividade principal da organização. São concebidos com a essência de um sistema e seu domínio da aplicação, e mudam mais lentamente que requisitos voláteis. Em geral, eles são derivados do Modelo de Domínio.

Eu particularmente não gosto muito desse nome e preferia que fosse chamado de *requisitos estáveis* em vez de *requisitos permanentes*, porque o termo *permanente* dá a impressão de que os requisitos não mudam e nós sabemos que requisitos sempre mudam. **Os requisitos permanentes são mais estáveis e que mudar pouco ou demoram bastante para mudar.** *Diego, você pode dar um exemplo de requisito permanente?* Claro, vamos lá...

Imaginem um domínio de aplicação (Ex: Bolsa de Valores). Ora, é natural que – em um sistema da Bolsa de Valores – existam sempre requisitos relacionados a ações, câmbio, cotações, índices, etc. Se, daqui vinte anos, um outro sistema for feito para a Bolsa de Valores, é bem provável que continue existindo requisitos relacionados a ações, câmbio, cotações, índices, etc. **Pode mudar uma coisa ou outra, mas esses requisitos são mais estáveis com o passar do tempo.**

- **Requisitos Voláteis:** também chamados de Requisitos Instáveis, são específicos para a instanciação de um sistema em um ambiente ou um cliente particular e são mais propensos a mudança. Se modificam quando o sistema está em desenvolvimento ou em uso. Podem ser subclassificados em mutáveis, emergentes, consequentes ou de compatibilidade:
  - **Mutáveis:** são os requisitos que se modificam em função de mudanças no ambiente no qual o sistema opera. Por exemplo, os requisitos para um sistema que calcula taxas de dedução que evoluem conforme as leis fiscais são atualizadas (muito comum no Brasil).
  - **Emergentes:** são os requisitos que não podem ser completamente definidos quando o sistema é especificado e emergem (olha a dica!) à medida que a compreensão do cliente sobre o sistema se desenvolve. Em geral, eles só aparecerão durante o desenvolvimento.
  - **Consequentes:** são os requisitos baseados em suposições de como o sistema será utilizado, isto é, são resultado da introdução do sistema no ambiente do usuário. O usuário percebe as necessidades enquanto utiliza sistema e esses requisitos são uma consequência do uso.
  - **De Compatibilidade:** são os requisitos que dependem de outro equipamento, processo, componente ou elemento. Conforme outros elementos mudam, esses requisitos também mudam. Esses são requisitos menos comuns, mas que também ocorrem.



Imaginem que, no desenvolvimento do sistema, deseja-se que o Índice IBOV da Bovespa fique posicionado no canto inferior direito da tela, mas depois eu percebo que esse é um dado muito importante para ficar isolado no canto, então eu mudo o meu requisito para que essa informação fique bem grande no centro da tela. *Vocês percebem como esse é um requisito mais instável?* **Outra pessoa pode achar que o índice não é tão importante para ficar no centro da tela.**

**(UNEAL – 2009)** São requisitos relativamente estáveis derivados da atividade central da organização e que se relacionam diretamente ao domínio do sistema. Qual opção abaixo corresponde à descrição anterior?

- a) Requisitos mutáveis
- b) Requisitos emergentes
- c) Requisitos conseqüentes
- d) Requisitos permanentes
- e) Requisitos de compatibilidade

**Comentários:** requisitos relativamente estáveis são os requisitos permanentes (Letra D).

**(SEFAZ/BA – 2014)** Segundo Sommerville (2004), os requisitos são divididos em duas classes. São elas:

- a) requisitos voláteis e requisitos funcionais.
- b) requisitos permanentes e requisitos voláteis.
- c) requisitos de compatibilidade e requisitos mutáveis.
- d) requisitos mutáveis e requisitos emergentes.
- e) requisitos emergentes e requisitos conseqüentes.

**Comentários:** os requisitos podem ser divididos em permanentes e voláteis (Letra B).

**(BAHIAGÁS – 2016 – Item IV)** Requisitos permanentes são requisitos que irão mudar durante o processo de desenvolvimento do sistema ou depois que o sistema estiver em operação.

**Comentários:** a questão trata dos requisitos voláteis (Errado).





## Classificação quanto à funcionalidade

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Percebam o seguinte: podemos classificar requisitos de diversas maneiras. No entanto, **não se discute que a classificação de requisitos quanto a sua funcionalidade é, com total e absoluta certeza, a classificação mais tradicional e mais frequentemente cobrada em provas**. Galera, se eu tivesse que apostar em algum assunto para cair na prova, eu apostaria nesse! No entanto, fiquem tranquilos porque é fácil ;)

- **Requisitos Funcionais:** são ações ou funcionalidades que o sistema deve fornecer para atingir seus objetivos. Eles dependem do tipo de software, dos usuários esperados e do tipo de sistema onde o software será implantado e fazem parte da arquitetura de um sistema. Grosso modo, pode-se dizer que eles tratam de **o que** o sistema deve fazer enquanto os requisitos não-funcionais tratam de **como** o sistema deve fazer.

### REQUISITOS FUNCIONAIS

Declarações de serviços que o sistema deve fornecer.

Como o sistema deve reagir a entradas específicas.

Como o sistema deve comportar em determinadas situações.

Podem estabelecer explicitamente o que o sistema não deve fazer.

Podem ser descritos como requisitos de usuário (abstrato).

Em regra, descrevem a função do sistema detalhadamente, com entradas, saídas, exceções, etc.

Em princípio, sua especificação deve ser completa e consistente.

**Problemas dos Requisitos Funcionais:** frequentemente, requisitos funcionais não são estabelecidos precisamente. Há descrições de requisitos ambíguos, que permitem diversas interpretações; há também requisitos incompletos, que não descrevem toda a funcionalidade do serviço; e há requisitos inconsistentes, que contradizem outros requisitos do sistema.

### Exemplos de Requisitos Funcionais:

- Pensemos em um Requisito do Outlook: **Sistema deverá fornecer** opção de filtrar e-mails por Assunto e Anexos.
- Pensemos em um Requisito do Youtube: **Sistema deverá reagir** com suspensão de vídeos que ferem direito autoral.
- Pensemos em um Requisito do GoogleMaps: **Sistema deverá procurar** rua mais próxima, caso não encontre a desejada.

**Requisitos Não-Funcionais:** são **restrições ou condições estipuladas sobre as quais o sistema deve funcionar**. Não estão diretamente relacionados às funções específicas do sistema, mas às gerais – e podem incluir restrições de tempo, restrições de processo de desenvolvimento, restrições impostas por padrões, entre outras. Podem ser mais críticos que os funcionais e sempre devem ser verificáveis. Eles fazem parte da arquitetura técnica de um sistema.



## REQUISITOS NÃO-FUNCIONAIS

Definem restrições globais e fazem parte da arquitetura técnica de um sistema.

Não se preocupam diretamente com a funcionalidade em si.

Colocam restrições sobre a qualidade e os atributos do sistema.

Incluem características como confiabilidade, segurança, usabilidade, performance, custos, robustez, etc.

**Problemas de Requisitos Não-Funcionais:** frequentemente, requisitos não-funcionais são bastante difíceis de se especificar objetivamente. Para tal, utilizam-se medidas que possam ser testadas ou mensuradas. No entanto, o problema mais comum são os requisitos conflitantes. Exemplo: o cliente deseja um desempenho altíssimo, mas quer que o sistema tenha baixo custo.

### Exemplos de Requisitos Não-Funcionais:

- Pensemos em um Requisito do Whatsapp: **Sistema deverá fornecer** disponibilidade mínima de 99,8%.
- Pensemos em um Requisito do Facebook: **Sistema deverá ser desenvolvido** utilizando a linguagem Java.
- Pensemos em um Requisito do Android: **Sistema deverá ser capaz** de rodar com apenas 1Gb de RAM.

---

**Requisitos de Domínio: são requisitos derivados do domínio da aplicação e refletem características de sua área de negócio.** Eles podem ser requisitos funcionais ou não-funcionais e, caso não sejam satisfeitos, o sistema pode não ser realizável. Por exemplo, um avião que não atende aos requisitos de confiabilidade, não será certificado para voo.

Pessoal, requisitos de domínio nada mais são que requisitos relacionados a um domínio de aplicação específico, sendo funcional ou não-funcional. **Sinceramente, galera... essa classificação não tem a menor utilidade.** A boa notícia é que ela não cai muito e que, na última edição do Sommerville, ele até retirou todo o destaque dessa classificação – colocando de forma bastante isolada no livro. Vamos seguir...

**Problemas de Requisitos de Domínio:** frequentemente, requisitos de domínio são descritos na linguagem ou jargão do domínio da aplicação. Especialistas de domínio compreendem tão profundamente assuntos sobre a sua área que eventualmente eles deixam de detalhar informações importantes, por acharem que são óbvias demais ou que já estão subentendidas.

### Exemplos de Requisitos de Domínio:

- Pensemos em um Requisito da USP: **Sistema deverá calcular** a nota final da prova de mestrado segundo a fórmula  $(1.74N_1 + 3.17N_2 + 2.59N_3)/3 * IRA$  (Aluno).
- Pensemos em um Requisito da STN: **Sistema deverá calcular** o valor do Título Público NTN-B multiplicado pelo valor da Taxa SELIC do mês corrente.



- Pensemos em um Requisito da NASA: **Sistema deverá funcionar** em uma Câmera 4K captando infravermelho e resistente a variações de temperatura.

---

De acordo com Sommerville: a distinção entre esses diferentes tipos de requisitos não é tão clara como sugere essas definições. Um requisito pode parecer-se inicialmente não funcional, mas quando desenvolvido com mais detalhes pode dar origem a uma série de novos requisitos funcionais. **Ao discutirmos sobre requisitos devemos levar em conta que na realidade a distinção entre eles é artificial.**

Nosso autor ainda diz que os requisitos não funcionais estão raramente associados às características individuais do sistema. Trata-se de qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras. Normalmente estes requisitos são descritos de maneira informal e, segundo ele, **geralmente afetam a arquitetura do sistema.** Fechado? A questão abaixo é o nosso desafio... tentem fazer sem olhar o gabarito!

**(SEFAZ/SP – 2013)** Dentre os requisitos obtidos para a construção do software constavam:

1. O software deve permitir as funções de cadastro, consultas diversas, alteração de dados e exclusão de alunos, professores e demais colaboradores.
2. O sistema deve ser fácil de usar, fácil de encontrar o que se procura e fácil de memorizar os passos para executar as operações mais comuns.
3. O sistema deve ter seu funcionamento baseado nas tecnologias web.
4. Todas as operações disponibilizadas no sistema devem contemplar a legislação vigente.
5. O sistema deve fazer interface com o sistema da Receita Federal por meio de requisições/respostas utilizando XML.
6. Os alunos devem poder obter por meio do sistema informações sobre suas faltas e notas em cada disciplina.
7. O boletim e o histórico do aluno poderão ser consultados e visualizados pelos gestores, funcionários da secretaria e pelo próprio aluno.
8. Ao clicar em uma opção para gerar o boletim do aluno, deve ser apresentada ao solicitante uma tabela com todas as disciplinas que o aluno cursou, bem como as notas das provas e o número total de faltas em cada disciplina.



9. O sistema deve responder à solicitação de geração do boletim de um aluno em no máximo 10 segundos.
  10. O sistema deve calcular a média aritmética das duas maiores dentre três notas de cada disciplina no final do semestre.
  11. Quando o sistema constatar que o aluno tem mais que 25% de faltas em uma disciplina do semestre, deve ser exibida no boletim do aluno a informação "Reprovado".
  12. O sistema deverá suportar a execução em qualquer plataforma de hardware e/ou sistema operacional.
  13. O sistema deve enviar automaticamente para o e-mail dos gestores autorizados um relatório com o número de alunos inadimplentes por curso.
  14. O sistema não deve revelar quaisquer dados pessoais dos alunos aos professores, exceto informações sobre notas e faltas no curso em que o professor leciona.
  15. O sistema deve permitir que o professor inclua ou modifique as notas de seus alunos durante o semestre letivo.
  16. A quantidade de memória necessária para que um terminal possa executar o sistema nas condições mínimas aceitáveis é de 1 gigabyte.
  17. A taxa aceitável de falhas nas operações realizadas pelo usuário no sistema deve ser de 1 falha para cada 200 operações.
  18. O sistema e sua respectiva documentação deverão ser entregues em um ano a partir da data atual.
  19. O sistema não deve permitir operações que beneficiem alguns usuários em detrimento de outros.
  20. A interface do usuário deve ser construída utilizando HTML5 e CSS.
  21. Se a média do aluno por disciplina, calculada no final do semestre, for menor do que 7, deve ser exibido no boletim do aluno a informação "Reprovado".
- Baseado nos requisitos apresentados, é correto afirmar que são requisitos funcionais os de números:

- a) 1, 2, 6, 10, 11, 14, 15, 16 e 21.
- b) 1, 6, 8, 10, 11, 13, 14, 17, 18 e 19.
- c) 1, 6, 7, 8, 10, 11, 13, 15 e 21.



- d) 1, 3, 4, 8, 10, 11, 12, 13, 15, 18 e 21.  
e) 2, 3, 4, 5, 9, 12, 14, 16, 17, 18, 19 e 20.

**Comentários:** Requisitos Funcionais (RF): 1, 6, 7, 8, 10, 11, 13, 15, 21; Requisitos Não-Funcionais (RNF): 2, 3, 4, 5, 9, 12, 14, 16, 17, 18, 19, 20. Não era necessário analisar todos os requisitos, era possível fazer por eliminação. *Como assim?* 1 é RF, elimina-se a Letra E; 2 é RNF, elimina-se a Letra A; 3 é RNF, elimina-se a Letra D; 7 é RF, elimina-se a Letra B (Letra C).

**(TRF4 – 2019)** Suponha que um Analista de TI, participando da etapa de análise de requisitos de um sistema de emissão de certidão negativa para o TRF4, tenha elencado os requisitos apresentados abaixo:

1. Utilizar interface responsiva para que possa ser executado em dispositivos móveis e na web.
2. Validar o tipo de certidão solicitado.
3. Emitir certidão negativa após verificação de situação do requerente.
4. Solicitar o CPF do requerente.
5. Responder ao clique único do usuário em qualquer botão da interface.
6. Validar o CPF do requerente.
7. Restaurar os dados automaticamente após falhas não programadas.
8. Solicitar o nome do requerente.
9. Oferecer dois tipos de certidão: para fins gerais e para fins eleitorais.
10. Emitir aviso de impossibilidade de emissão da certidão.

Sobre os requisitos, é correto afirmar que

- a) todos são funcionais.
- b) todos são não funcionais.
- c) 1, 5 e 7 são não funcionais.
- d) apenas 3, 4, 8, 9 e 10 são funcionais.
- e) apenas 2, 6 e 7 são não funcionais.

**Comentários:** Requisitos Funcionais (RF): 2, 3, 4, 6, 8, 9, 10; e Requisitos Não-Funcionais (RNF): 1, 5 e 7 (Letra C).

**(STJ – 2018)** Os requisitos funcionais especificam o que o software deverá fazer. Esses requisitos incluem tempo de resposta, utilização de volumetria estática, escalabilidade, disponibilidade, segurança e usabilidade.

**Comentários:** a primeira parte da questão está perfeita! No entanto, são os requisitos não-funcionais que incluem tempo de resposta, utilização de volumetria estática, escalabilidade, disponibilidade, segurança e usabilidade (Errado).



## Classificação quanto à origem

INCIDÊNCIA EM PROVA: MÉDIA

Agora vamos partir para outra classificação um pouquinho menos importante, mas que – vez ou outra – cai em prova! Pesquisadores observaram que **os requisitos não-funcionais também podiam ser agrupados por meio de suas características comuns**. Para tanto, criou-se a subclassificação dos requisitos não-funcionais em: requisitos de produto, requisitos organizacionais e requisitos externos.

- **Requisitos de Produto:** especificam o comportamento do produto. Entre os exemplos, estão requisitos de desempenho quanto à rapidez com que o sistema deve operar e quanto de memória ele requer, requisitos de confiabilidade que definem a taxa aceitável de falhas, requisitos de portabilidade e requisitos de usabilidade.
- **Requisitos Organizacionais:** são derivados de políticas e procedimentos da organização do cliente e do desenvolvedor. Entre os exemplos, estão padrões de processo que devem ser usados, linguagem de programação ou o método de projeto usado, e requisitos de entrega que especificam quando o produto e a sua documentação devem ser entregues.
- **Requisitos Externos:** abrange todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento. Entre os exemplos, estão a interoperabilidade que define como o sistema interage com outros sistemas, requisitos legais que devem ser seguidos, requisitos éticos sistema para assegurar que ele será aceito por todos.

### IMPORTANTE

A INTEROPERABILIDADE É UM REQUISITO DE PRODUTO, ORGANIZACIONAL OU EXTERNO? ESTOU CITANDO ESSA FUNCIONALIDADE ESPECÍFICA, PORQUE JÁ VI DIVERSAS PESSOAS PENSANDO QUE É DE PRODUTO. NA VERDADE, ELA É UM REQUISITO EXTERNO, PORQUE DEPENDE DE PADRONIZAÇÕES FORA DE SEU CONTROLE.

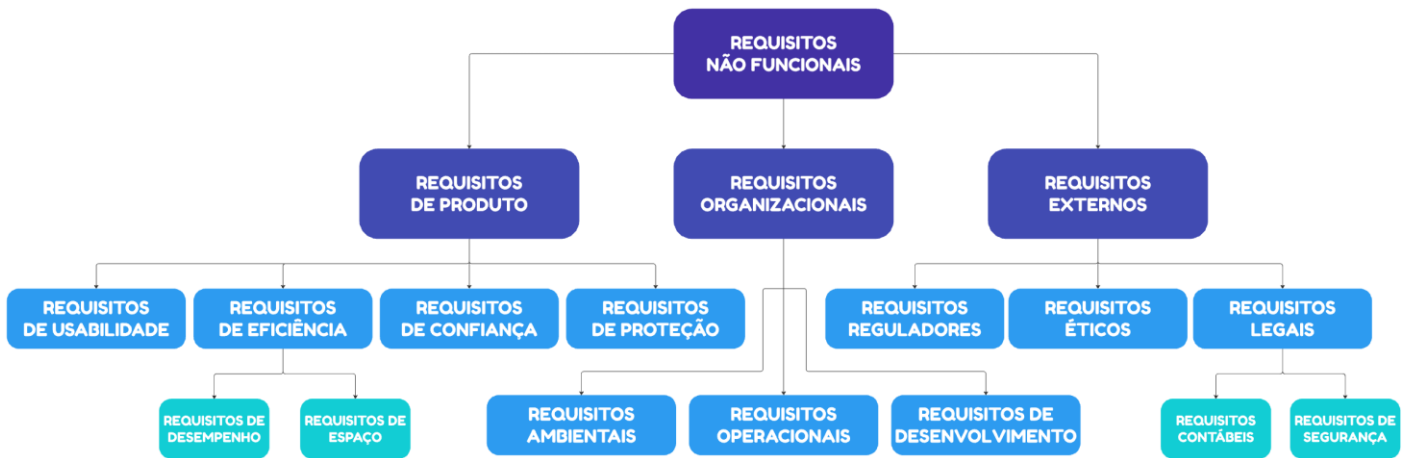
*Acabaram as classificações, professor? Infelizmente, não! Cada uma dessas três subclassificações se divide também em várias outras, como mostra a imagem seguinte.* Cabe salientar que essa classificação está em conformidade com a 9ª Edição do livro do Ian Sommerville, em que há pequenas diferenças quanto à mesma imagem de edições anteriores. Vamos lá...

REQUISITOS	EXEMPLOS
REQUISITOS DE CONFIABILIDADE	O sistema não deve ficar fora do ar por mais de cinco segundos durante o dia.
REQUISITOS DE PROTEÇÃO	O sistema não deve permitir que os usuários modifiquem senhas de acesso que eles não criaram.





<b>REQUISITOS DE DESEMPENHO</b>	O sistema deverá ser capaz de processar oitocentas requisições por segundo.
<b>REQUISITOS DE ESPAÇO</b>	Também chamado de Requisitos de Armazenamento, o sistema deverá ocupar, no máximo, 80Mb da memória interna do dispositivo.
<b>REQUISITOS DE USABILIDADE</b>	Os usuários deverão operar todas as funcionalidades do sistema após 2 horas de treino.
<b>REQUISITOS DE SEGURANÇA</b>	O sistema não deve permitir a ativação simultânea de mais de três sinais de alarme.
<b>REQUISITOS ÉTICOS</b>	O sistema não apresentará aos usuários quaisquer dados de natureza confidencial de outrem.
<b>REQUISITOS DE IMPLEMENTAÇÃO</b>	A interface de usuário deve ser implementada em HTML e não se deve utilizar Applets de Java.



**(MPE/MA – 2013)** O escopo de um projeto é determinado pelo levantamento de requisitos funcionais e não funcionais. Dentre os requisitos não funcionais se enquadram os requisitos organizacionais, que podem ser divididos em:

- a) reguladores e éticos.
- b) ambientais, operacionais e de desenvolvimento.
- c) contábeis e de segurança.
- d) de desempenho e de espaço.
- e) de eficiência, de confiança e de proteção.

**Comentários:** os requisitos organizacionais se dividem em ambientais, operacionais e de implementação (ou desenvolvimento) (Letra B).



## Engenharia de Requisitos

INCIDÊNCIA EM PROVA: ALTÍSSIMA

Vamos começar pelo histórico! **O termo *engenharia de software* foi utilizado pela primeira vez na década de 70 em um relatório técnico da TRW Inc., mas não chegou a se tornar um jargão.** Ele só ficou de fato mais conhecido na década de noventa com a publicação de um tutorial da IEEE e a criação de uma série de conferências sobre esse assunto. *Ok, professor... mas qual é a definição de Engenharia de Requisitos?*

Bem, ela pode ser definida como **uma abordagem sistemática para a formulação, análise, documentação e manutenção de requisitos de um sistema.** Em outras palavras, também podemos defini-la como um processo formal que engloba todas as atividades que contribuem para a produção de um documento de requisitos. Vocês percebem que são duas visões um pouco diferentes de engenharia de requisitos, mas que elas possuem um detalhe em comum. *Qual é?*

A primeira trata de uma abordagem **sistemática** e a segunda trata de um processo **formal**. *Pessoal, porque essas são as palavras-chave da definição?* Porque sempre que a palavra engenharia aparece, você já pode considerar que se trata de uma abordagem ou processo formal e sistemático. Então se uma prova discursiva te pergunta o que é Engenharia de Requisitos, você já sabe que se trata de algo formal, metodológico, sistemático, processual, repetível, etc.

Vou provar para vocês! *O que é a Engenharia Civil?* É o **processo formal** para concepção, projeto, construção e manutenção de diversas infraestruturas. A Engenharia Mecânica é o **processo formal** para a concepção, análise, fabricação e manutenção de sistemas mecânicos. *Logo, qual é a definição de Engenharia de Software?* É o **processo formal** para o levantamento, análise, especificação, validação, gerenciamento e manutenção de requisitos.

*E o que essas definições querem dizer com "processo formal"?* **Elas querem dizer que se trata de uma sequência de passos repetíveis e documentados** de modo que seja possível construir um carro no México ou na Holanda que o resultado será semelhante. Vindo para o mundo da engenharia de requisitos, isso significa que duas empresas diferentes podem levantar os requisitos de um sistema de software que provavelmente obterão resultados semelhantes.

*Galera, qual é a importância da engenharia de requisitos?* **Bem, essa é a fase mais crítica no desenvolvimento de um software**, tendo em vista que erros durante esse estágio conduzem inevitavelmente a problemas posteriores no projeto e na implementação do sistema. *Professor, isso não é simples de resolver? Basta eu contratar um bom engenheiro de requisitos, ele vai até a organização, levanta os requisitos corretos e fim – problema resolvido.*

**Galera, quem dera se fosse assim... a verdade é que tudo bem mais complicado do que parece.** Por essa razão, nós temos a disciplina de engenharia de requisitos: para que ela nos oriente sobre como formular, elicitar, analisar, documentar, manter e gerir os requisitos de um sistema de

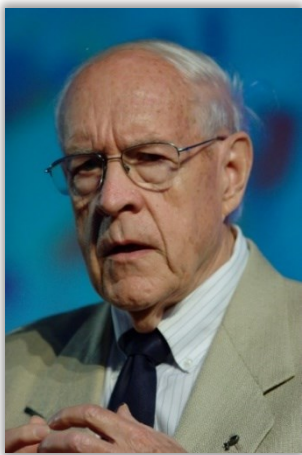


software. Quem já trabalhou com isso sabe que às vezes o usuário pede uma coisa, mas na verdade ele quer outra coisa completamente diferente do que ele disse.

**Sim, pessoal! Há usuários que não sabem expressar muito bem em palavras o que ele deseja.** Em outras ocasiões, um mesmo usuário pede dois requisitos totalmente contraditórios sem perceber. Há também o conflito de requisitos entre usuários diferentes de uma mesma empresa, uma vez que – em uma organização – podem existir diferentes visões sobre um mesmo produto de software. Galera, rola umas tretas sinistras por conta disso...

Acontece também de a organização não ter um processo de trabalho muito maduro e implantado causando problemas sérios no levantamento de requisitos, uma vez que os requisitos acabam sendo modificados durante o próprio levantamento ou a especificação dos requisitos. Então, **a engenharia de requisitos nos traz ferramentas e técnicas para ajudar a mitigar esses problemas, mas tenham certeza: mesmo com tudo que isso, ela não consegue resolver todos os problemas.**

Por outro lado, **um sólido processo de engenharia de requisitos é capaz de encontrar a melhor solução viável no momento.** Na prática, é praticamente impossível satisfazer absolutamente todas as expectativas dos clientes, na medida em que requisitos podem ser bastante complexos e voláteis. Lembrem-se que requisitos são entidades vivas que vão mudando o tempo todo por diversos motivos – a engenharia de requisitos faz o melhor que ela pode!



**Na prática, os requisitos de sistema se modificam.** Isso ocorre, porque as pessoas envolvidas desenvolvem uma compreensão maior do que desejam que o software faça; a própria organização que está comprando o sistema muda; modificações são feitas no hardware, software e no ambiente organizacional do sistema; entre outros.

**O Engenheiro de Software da IBM, Fred Brooks, diz:** *“A parte mais árdua na construção de um sistema de software consiste em decidir o que construir! Nenhuma outra fase do processo de desenvolvimento compromete tanto o resultado final do sistema se feita errada e nenhuma outra dificulta tanto as correções posteriores”.*

#### SEGUNDO MIRANDA (2002 APUD SANTOS, 2007, P.12):

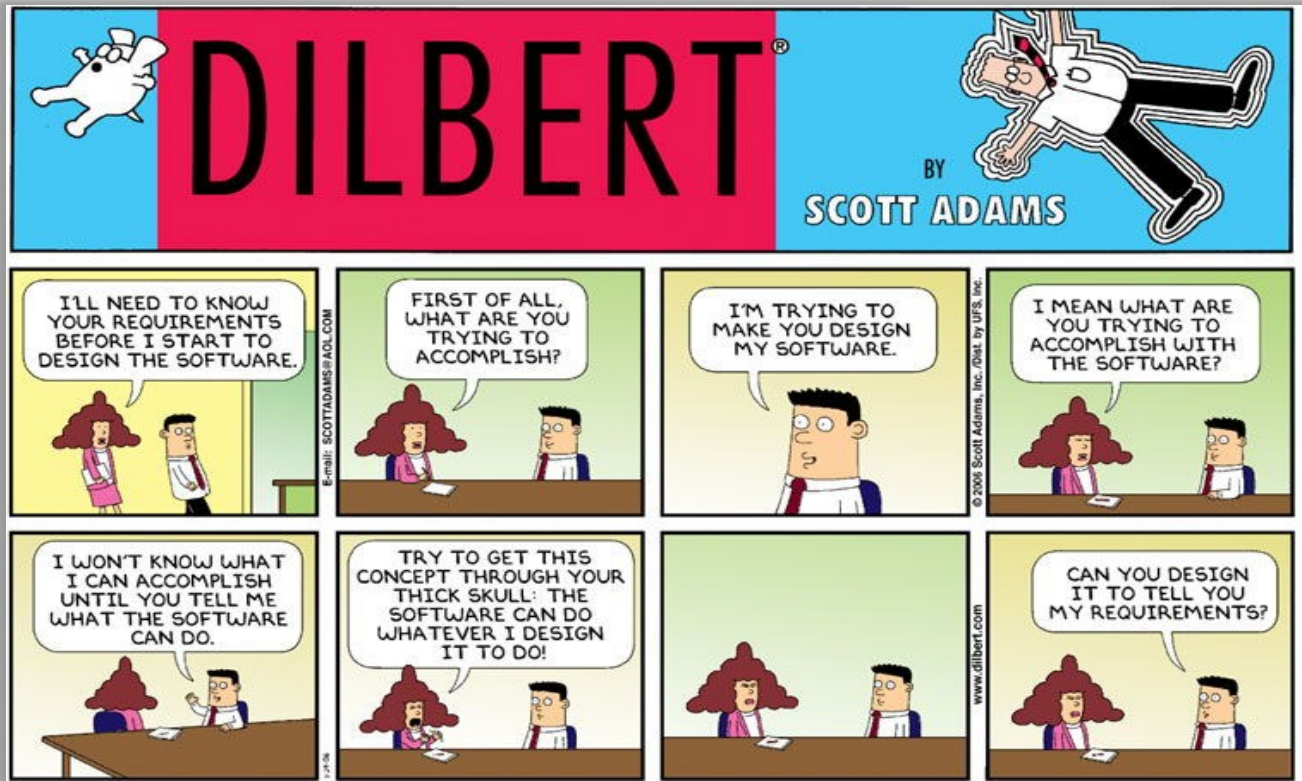
50% dos principais defeitos de software são oriundos da fase de especificação de software.

12% das principais causas de fracassos em projetos são oriundos de requisitos incompletos.

12% das principais causas de sucesso em projetos são oriundos de requisitos consistentes.

De modo geral, a engenharia de requisitos fornece um mecanismo adequado para compreender o que o cliente deseja, para analisar as suas necessidades, para avaliar a viabilidade da implementação, para negociar uma solução razoável com os usuários, para especificar uma solução não-ambígua, para validar a especificação e para gerenciar os requisitos **à medida que eles são transformados em um sistema operacionalizável.**





Enfim, chegamos à nossa última pergunta! *Quais são as fases do processo de Engenharia de Requisitos?* Bem, há duas respostas para essa pergunta! *Como assim, professor?* É que há uma divergência entre os principais autores. Nós vamos nos aprofundar nas fases do Sommerville, **porque ele é o autor mais cobrado em provas nesse quesito**, porém não custa nada ver também as fases do Pressman abaixo:

FASES	DESCRIÇÃO
CONCEPÇÃO	Após uma necessidade de o negócio ser identificada, busca-se estabelecer um entendimento básico do problema. Trata-se da concepção inicial do software e busca entender o problema, quem são os envolvidos, a natureza da solução e iniciar o processo de comunicação entre clientes e colaboradores.
LEVANTAMENTO	Etapa crítica, utiliza uma abordagem organizada para descobrir o que o cliente deseja em seu sistema. Envolve intensa participação do stakeholders e faz três perguntas: Qual o objetivo do produto? Como o produto se enquadra nas necessidades do negócio? Como o produto será utilizado?
ELABORAÇÃO	Por vezes chamada Análise, informações obtidas do cliente durante a concepção e levantamento são expandidas e refinadas em um modelo, definindo o domínio do problema. Incluem-se modelagens de cenários de interação do usuário com o sistema e modelagens das classes envolvidas.
NEGOCIAÇÃO	Tem por objetivo chegar a um consenso sobre os conflitos entre clientes e usuários, por intermédio de um processo de negociação. Os requisitos são avaliados junto ao cliente e podem se combinar, excluir ou até mesmo inserir novos requisitos.
ESPECIFICAÇÃO	Por vezes chamada Documentação, produto final do engenheiro de requisitos, pode ser um documento escrito, um modelo gráfico, cenários de uso, protótipos, etc. Trata-se da





	apresentação formal dos dados obtidos até o momento de modo que possa guiar o desenvolvimento futuro do software.
<b>VALIDAÇÃO</b>	Os produtos de trabalho resultantes da engenharia de requisitos são avaliados quanto a sua qualidade por todos os envolvidos (clientes, colaboradores e usuários). Buscam-se erros de interpretação, ambiguidades e omissões.
<b>GESTÃO</b>	conjunto de atividades que auxiliam a equipe de projeto a identificar, controlar e rastrear requisitos e mudanças nos requisitos a qualquer momento. Para projetos de grande porte, é uma fase essencial na medida em que mudanças em um requisito podem afetar diversos outros requisitos.

**Em resumo:** na fase de *concepção*, tenta-se entender o problema a ser resolvido e cria-se uma concepção do software que resolverá esse problema; na fase de *levantamento de requisitos*, busca-se descobrir os requisitos que o sistema terá; na *fase de elaboração*, também chamada de análise de requisitos, busca-se expandir e refinar esses requisitos – adicionando mais detalhes; na fase de negociação, busca-se um consenso sobre requisitos contraditórios, diferentes, etc.

Já na fase de *especificação de requisitos*, também chamada de *documentação de requisitos*, cria-se o documento de requisitos; na fase de *validação de requisitos*, realiza-se uma revisão do documento de requisitos e avalia-se a sua qualidade; e, por fim, na fase de *gestão*, busca-se manter um controle e rastreabilidade dos requisitos do sistema de software. **Se vocês compreenderam tudo isso, podemos seguir para as fases de acordo com o Sommerville!**

POR SOMMERVILLE		POR PRESSMAN
ESTUDO DE VIABILIDADE	→	CONCEPÇÃO
ELICITAÇÃO E ANÁLISE DE REQUISITOS	→	LEVANTAMENTO
OBTENÇÃO DE REQUISITOS	→	ELABORAÇÃO
CLASSIFICAÇÃO E ORGANIZAÇÃO	→	
PRIORIZAÇÃO E NEGOCIAÇÃO	→	NEGOCIAÇÃO
DOCUMENTAÇÃO DE REQUISITOS	→	
ESPECIFICAÇÃO	→	ESPECIFICAÇÃO
VALIDAÇÃO	→	VALIDAÇÃO
GESTÃO	→	GESTÃO

Na tabela acima, podemos ver uma tabela comparativa entre as fases de acordo com cada autor. *Como decorar, professor?* Bem, chegou a hora de eu ganhar alguns inimigos! Quando eu estudava esse assunto, eu criei mnemônicos para decorar as fases de cada autor. Eu já levei um sapo de um aluno no fórum de dúvidas por usar o Vasco como exemplo - Vascaínos, peço que não se sintam ofendidos. **Mnemônicos devem ser absurdos mesmo para ajudar na memorização, ok?**





Onefootball  
Alexandre Fernandes · 10 de out de 2019

## Cenas lamentáveis em novo empate do Vasco da Gama

### FASES DE ACORDO COM IAN SOMMERVILLE

ESTÃO VIABILIZANDO ELIANA ESPECIALMENTE NO VASCO DA GAMA

ESTUDO DE VIABILIDADE > ELICITAÇÃO E ANÁLISE DE REQUISITOS > ESPECIFICAÇÃO > VALIDAÇÃO > GESTÃO

### FASES DE ACORDO COM ROGER PRESSMAN

CENAS LAMENTÁVEIS EM NOVO EMPATE DO VASCO DA GAMA

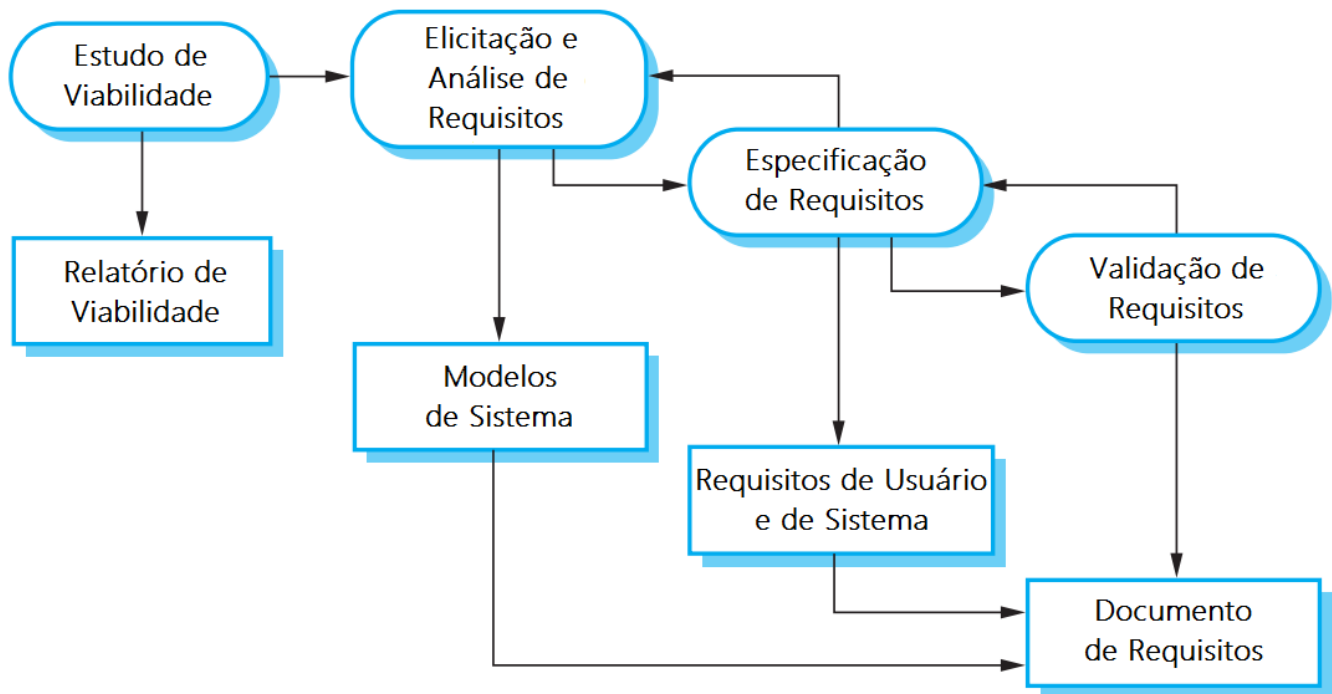
CONCEPÇÃO > LEVANTAMENTO > ELABORAÇÃO > NEGOCIAÇÃO > ESPECIFICAÇÃO > VALIDAÇÃO > GESTÃO

A imagem a seguir apresenta graficamente as fases da Engenharia de Requisitos de acordo com Sommerville. São elas: Estudo de Viabilidade, Elicitação e Análise de Requisitos, Especificação de Requisitos e Validação de Requisitos. Por fim, há uma última fase que não está representada na imagem, mas que envolve e suporta todas as fases anteriores e também é de suma importância: Gestão/Gerenciamento de Requisitos.

Sommerville afirma que o objetivo da engenharia de requisitos é criar e manter um documento de requisitos de sistema. Assim, note que o resultado do *Estudo de Viabilidade* é o *Relatório de Viabilidade*; o resultado da *Elicitação e Análise de Requisitos* é um conjunto de *Modelos do Sistema*; o resultado da *Especificação de Requisitos* são os *Requisitos de Usuário e de Sistema*; e o resultado da *Validação de Requisitos* é o famoso *Documento de Requisitos*.







**Todos esses artefatos servem de insumo para construir o Documento de Requisitos!** Além disso, percebam que as setas vão e voltam entre as fases. Então, no momento de validar os requisitos, por exemplo, se eu encontrar algo incorreto, eu posso retornar à *Especificação de Requisitos*. Bacana? Então, agora é o momento de entrar em mais detalhes sobre cada uma dessas fases que nós acabamos de ver. Vem comigo...

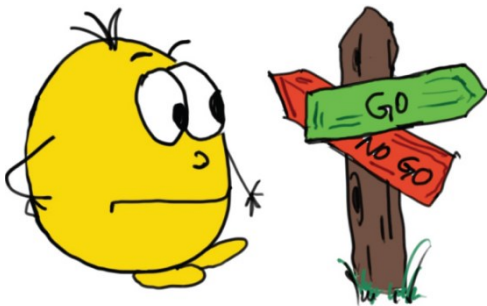


## Estudo de Viabilidade

A fase de Estudo de Viabilidade trata da realização de uma avaliação relativamente rápida e barata para verificar se as necessidades identificadas dos usuários podem ser satisfeitas por meio das tecnologias atuais de sistemas de software e hardware. **O resultado dessa avaliação deve fornecer informações para que a alta direção da organização tome uma decisão mais embasada** quanto a prosseguir para uma análise mais detalhada ou não.

**Esse é o momento de fazer diversos questionamentos importantes:** *o sistema realmente agregará valor ao negócio? Ele será útil para a empresa? Ele será rentável? Qual é o retorno de investimento que ele será capaz de realizar? É viável tecnologicamente e financeiramente?* Imaginem um sistema de software que automatiza algum processo de negócio de uma organização, mas que custa R\$10 milhões para implementar e economiza apenas R\$5.000/mês.

De uma outra forma, pode ser descrita como atividade inicial do processo de engenharia de requisitos, consistindo em um conjunto preliminar de requisitos de negócio, um esboço da descrição do sistema e da forma como o sistema pretende apoiar os processos de negócios da organização. **A fase seguinte utiliza as informações do estudo de viabilidade como base para o levantamento de requisitos.**



Além disso, **ele deve ser um estudo curto, bem focado e preferencialmente barato**, devendo ser realizado no início do processo de engenharia de requisitos e, ao final, deverá entregar um relatório de viabilidade. *Por que?* Porque se você perder muito tempo desenvolvendo o estudo ou se ele for caro demais, pode acabar não valendo a pena sequer fazê-lo. Com o resultado do estudo em mãos, a alta direção pode dar o famoso *go/no-go* (*vamos ou não vamos seguir?*).

Por fim, o Estudo de Viabilidade deve responder três questões em que – **caso alguma delas tenha uma resposta negativa – o projeto não deve seguir adiante.** São elas:

1. O SISTEMA CONTRIBUI PARA OS OBJETIVOS GERAIS DA ORGANIZAÇÃO?
2. O SISTEMA PODE SER IMPLEMENTADO COM TECNOLOGIA ATUAL E DENTRO DO CUSTO E PRAZO?
3. O SISTEMA PODE SER INTEGRADO A OUTROS SISTEMAS JÁ IMPLANTADOS?

**As respostas para essas três questões não é algo tão simples de ser obtido, visto que geralmente a empresa não tem a exata definição de seus objetivos.** Para ajudar nesta definição, questões podem ser levantadas às partes interessadas e, com as respostas em mãos, pode-se concluir se o desenvolvimento do sistema deve prosseguir ou não; pode-se propor alterações de escopo, orçamento, prazo; etc. Segue uma lista de questionamentos possíveis:

1. SE O SISTEMA NÃO FOSSE IMPLEMENTADO, QUAL SERIA O RUMO DA ORGANIZAÇÃO?
2. QUAIS SÃO AS FALHAS DOS PROCESSOS ATUAIS E QUAIS SOLUÇÕES O NOVO SISTEMA TRARIA?



3. QUANTO AOS OBJETIVOS E REQUISITOS, QUAL A CONTRIBUIÇÃO DIRETA DO NOVO SISTEMA?
4. EXISTE A POSSIBILIDADE DE INTEGRAÇÃO COM OUTROS SISTEMAS?
5. QUAIS DAS TECNOLOGIAS SERÃO NOVAS?
6. O QUE O NOVO SISTEMA IRÁ APOIAR?

**(FUB – 2011)** O estudo de viabilidade, uma atividade inicial do processo de engenharia de requisitos, consiste em um conjunto preliminar de requisitos de negócio, um esboço da descrição do sistema e da forma como o sistema pretende apoiar os processos de negócios.

**Comentários:** em todos os sistemas novos, o processo de engenharia de requisitos deve começar com um estudo de viabilidade. A entrada para o estudo de viabilidade consiste em um conjunto preliminar de requisitos de negócios, um esboço da descrição do sistema e como o sistema pretende apoiar os processos de negócios (Correto).

**(CHESF – 2012)** O processo de engenharia de requisitos engloba todas as atividades necessárias para criar e manter um documento de requisitos do sistema e compreender os elementos de negócio que serão atendidos pelo software a ser desenvolvido, pertencendo a uma sequência lógica de atividades que culminam em um documento de requisitos correto que inclui todas as necessidades do cliente.

O primeiro passo necessário no processo de criação desse documento de requisitos deve ser a(o):

- a) análise de viabilidade
- b) prototipagem do sistema
- c) geração de casos de teste
- d) design da aplicação
- e) documento de gerenciamento de mudanças

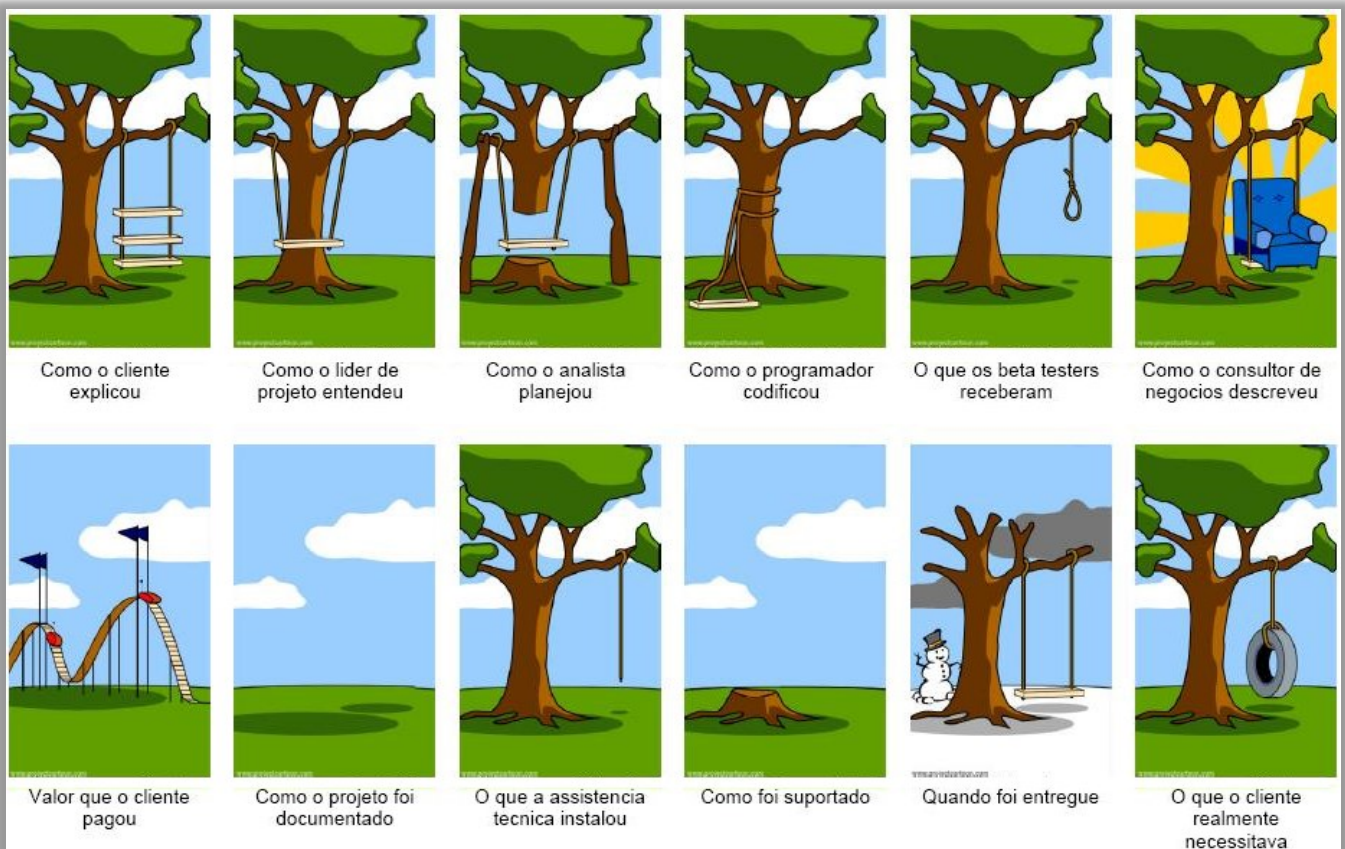
**Comentários:** em todos os sistemas novos, o processo de engenharia de requisitos deve começar com um estudo de viabilidade (Letra A).



## Elicitação e Análise de Requisitos

O verbo **Elicitar** não é muito utilizado em nosso cotidiano, mas significa o mesmo que **descobrir, identificar, deduzir, extrair, evocar, obter informações sobre uma questão específica**. Para tal, os engenheiros de software trabalham com os clientes e usuários finais do sistema para aprender sobre o domínio da aplicação, quais serviços o sistema deve fornecer, o desempenho esperado, restrições de hardware, entre outros quesitos.

A fase de **Elicitação e Análise de Requisitos** trata do **processo de levantamento e derivação de requisitos de sistema** através da observação de sistemas existentes, discussões com usuários e compradores potenciais, análise de tarefas, entre outros. Isso pode envolver o desenvolvimento de um ou mais modelos de sistema e protótipos, que ajudam o analista a compreender o sistema a ser especificado.



*Pessoal, por que é tão difícil levantar requisitos? Nós já discutimos um pouco sobre isso, mas agora nós vamos ver em detalhes. A imagem acima é uma das figuras mais clássicas da engenharia de software. O que ela quer dizer? É o seguinte: o cliente explica uma coisa, o líder de projeto entende outra, o analista planeja outra, o programador codifica outra, os beta testers testam outra e o consultor de negócios descreve outra também totalmente diferente.*

Não acabou! Aí o cliente acaba pagando muito mais caro do que devia, o projeto não é bem documentado, a assistência técnica instala errado, não há suporte, demora mais que o programado



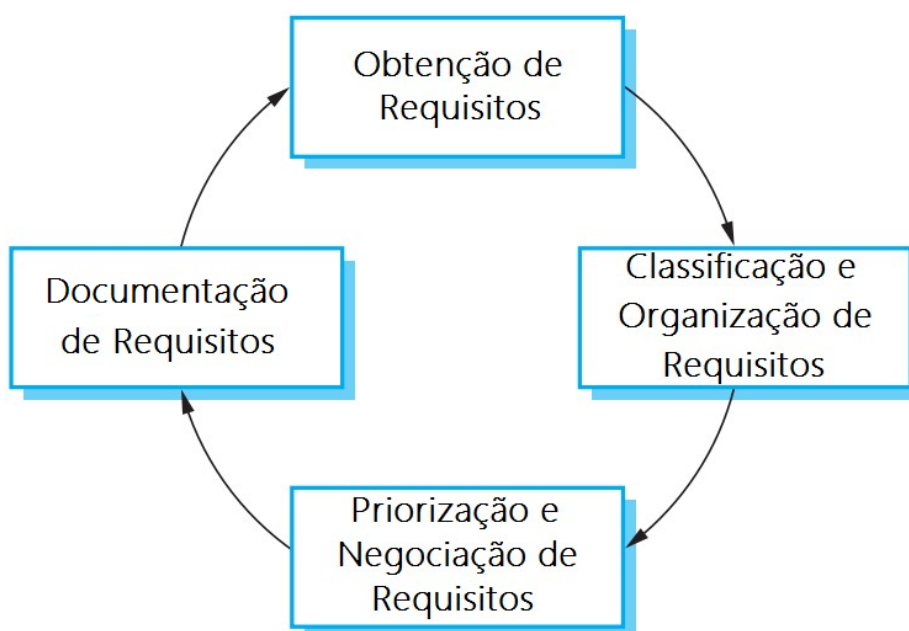
e, no fim, já estamos em outra estação do ano quando o sistema é entregue. Observem no último quadrinho que o cliente também não descreveu bem o que ele queria – **e é por isso que a engenharia de requisitos e suas técnicas são tão importantes.**



Galera, quem já trabalhou com isso sabe! Às vezes, os próprios clientes não sabem o que querem; às vezes, eles querem automatizar um processo que ainda não está efetivamente maduro; às vezes, eles esquecem de mencionar um requisito bastante importante porque para eles era óbvio para a área de tecnologia não era. Enfim... **a engenharia de requisitos está aqui para nos ajudar** a descobrir requisitos implícitos, contraditórios, inconsistentes, incompletos, entre outros.

As principais atividades do processo de *elicitação e análise de requisitos* são:

- a. **Obtenção de Requisitos:** processo de interação com os stakeholders para coletar requisitos. Os requisitos de domínio também são descobertos durante essa atividade.
- b. **Classificação e organização de requisitos:** esta atividade envolve a coleção de requisitos não estruturados, agrupa os requisitos relacionados e os organiza em conjuntos coerentes.
- c. **Priorização e negociação de requisitos:** inevitavelmente, os requisitos serão conflitantes. Assim, busca-se priorizar os requisitos e resolver conflitos por meio da negociação.
- d. **Documentação de requisitos:** os requisitos são documentados e colocados na próxima volta da espiral. Podem ser produzidos documentos de requisitos formais ou informais.





Em suma: (1) **Obtenção/Descoberta de Requisitos** – atividade de interação com as partes interessadas justamente para obter, descobrir e coletar os requisitos do sistema; (2) **Classificação e Organização de Requisitos** – momento de organizar os requisitos – eventualmente existe mais de um engenheiro de requisitos trabalhando em áreas diferentes de uma organização e, ao fim, eles apresentam seus requisitos, logo é preciso organizá-los.

*Vocês se lembram que nós vimos várias classificações de requisitos?* Pois é, os engenheiros de requisitos podem decidir classificá-los em requisitos funcionais ou não funcionais, em requisitos permanentes ou voláteis, em requisitos normais, esperados ou fascinantes, em requisitos de usuário ou de sistema, entre outras classificações. **O importante é classificar os requisitos, agrupá-los de acordo com algum critério e organizá-los em conjuntos que façam sentido.**

Além disso, nessa atividade serão realizadas algumas verificações preliminares de conflito, consistência, omissão e ambiguidade de requisitos. (3) **Priorização e Negociação de Requisitos** – como eu mencionei, são vários engenheiros de requisitos levantando requisitos com várias pessoas diferentes de uma organização e essas pessoas podem ter visões diferentes do produto, então é comum que os requisitos acabem entrando em conflito.

**Essa atividade busca consolidar as visões diferentes dos clientes em uma reunião e fazer com que elas conversem e cheguem a um consenso sobre os requisitos.** Então, esse é o momento de priorizar alguns requisitos em detrimento de outros e também de negociar para chegar a um acordo. *Bacana?* Saibam – de antemão – que é impossível satisfazer todos os stakeholders. Logo, não tem jeito, sempre haverá alguém insatisfeito.

(4) **Documentação de Requisitos** – trata-se da atividade de organizar os requisitos em um documento de maneira organizada, classificada, priorizada e negociada. Essa documentação é voltada, em termos técnicos para os engenheiros de requisitos e programadores e, em termos de negócio, para clientes e stakeholders. Bem, a fase de elicitação e análise de requisitos é campeã em frequência nas provas, principalmente a atividade de descoberta.

**(TRT/19 – 2011)** De acordo com Sommerville, são atividades do processo de elicitação de requisitos, pela ordem:

- a) casos de uso; análise; projeto; arquitetura.
- b) etnografia; casos de uso; análise; validação; arquitetura.
- c) entrevista; etnografia; documentação; registro.
- d) cenários; classificação; organização; priorização; documentação.
- e) obtenção; classificação e organização; priorização e negociação; documentação

**Comentários:** trata-se da obtenção, classificação e organização, priorização e negociação, e documentação (Letra E).

Para auxiliar a **assegurar uma cobertura ampla dos requisitos de um sistema de software**, utilizam-se as seguintes técnicas (as sublinhadas são as que caem mais em prova):





## 1. Entrevistas

Entrevistas formais ou informais com os stakeholders do sistema fazem parte da maioria dos processos de engenharia de requisitos. Nessas entrevistas, a equipe de engenharia de requisitos formula questões para os *stakeholders* sobre o sistema que eles usam e o sistema a ser desenvolvido. **Os requisitos são derivados a partir das respostas dessas questões.** As entrevistas podem ser abertas ou fechadas.

As entrevistas abertas ocorrem quando não há um roteiro predefinido e as entrevistas fechadas ocorrem quando há um roteiro predefinido. **Na prática, as entrevistas com os stakeholders são, geralmente, uma combinação desses tipos.** As respostas a algumas perguntas podem levar a outros assuntos discutidos de maneira menos estruturada. As discussões completamente abertas raramente funcionam bem.

Em regra, a maioria das entrevistas requer algumas perguntas como ponto de partida e para manter o foco no sistema a ser desenvolvido. **As entrevistas são úteis para obter um entendimento geral sobre o que os stakeholders fazem, como eles podem interagir com o sistema e as dificuldades que enfrentam com os sistemas atuais.** As pessoas gostam de falar sobre seu trabalho e, normalmente, ficam felizes em participar de entrevistas.

No entanto, as entrevistas não são tão úteis para compreender os requisitos do domínio da aplicação. Não é eficiente para eliciação de conhecimentos sobre os requisitos e as restrições organizacionais, pois existem relacionamentos sutis de poder e influência entre os stakeholders. **Em geral, a maioria das pessoas é relutante em discutir questões políticas e organizacionais que podem afetar os requisitos.**

**Por fim, existem três tipos de Entrevistas Formais: Pirâmide, Funil e Diamante.** A primeira começa com perguntas mais detalhadas e termina com questões mais genéricas; a segunda começa com perguntas mais genéricas e termina com perguntas mais detalhadas; e a terceira que é mescla de ambos –começa com perguntas mais detalhadas, depois são feitas perguntas mais genéricas e termina com perguntas mais detalhadas novamente.

## 2. Etnografia

**Técnica de observação utilizada para compreender os requisitos organizacionais e sociais.** Coloca-se o analista dentro do campo de atuação dos usuários, observando o trabalho diário anotando as tarefas reais em que os participantes estão envolvidos. Em geral, essa é uma técnica utilizada em conjunto com outras técnicas. Como ela é uma técnica de observação, isoladamente ela não é muito eficaz na eliciação.

O valor da técnica de etnografia está na ajuda que presta aos analistas em descobrir os requisitos implícitos de sistema que refletem os processos reais, e não os processos formais, com os quais as pessoas estão realmente envolvidas. **As pessoas frequentemente consideram muito difícil**



articular detalhes de seu trabalho rotineiro, visto que muitos passos são secundários e irrelevantes para elas.

É claro que elas compreendem seu próprio trabalho, mas podem não compreender seu relacionamento com o trabalho de outros na organização. Os fatores sociais e organizacionais, que afetam o trabalho, mas que não são claros e óbvios para as pessoas, podem somente se tornar claros quando examinados por um observador imparcial – por isso, a etnografia é importante!

### 3. Cenários

As pessoas geralmente consideram mais fácil relatar exemplos da vida real do que abstrair descrições. **Elas podem compreender e criticar um cenário de como interagiriam com um sistema de software.** Os engenheiros de requisitos podem usar as informações obtidas nessa discussão para elaborar os requisitos reais do sistema de software. Os cenários podem ser particularmente úteis para adicionar detalhes a um esboço da descrição de requisitos.

Eles são descrições de exemplos das sessões de interação. Cada cenário abrange uma ou mais interações possíveis. Diversos tipos de cenários foram desenvolvidos, cada um dos quais fornecendo diferentes tipos de informações sobre o sistema em diferentes níveis de detalhamento. **O uso de cenários para descrever requisitos é parte integrante dos métodos ágeis, como a Extreme Programming.**

O cenário começa com um esboço da interação e, durante a elicitação, os detalhes são adicionados para criar uma descrição completa dessa interação. A elicitação baseada em cenários pode ser realizada também de informalmente. Os engenheiros de requisitos trabalham com os stakeholders para identificar cenários e captar seus detalhes em forma de textos, diagramas, imagens, eventos, casos de uso, etc.

### 4. Questionários

**Formulários distribuídos aos stakeholders com questões pré-definidas. Torna-se útil quando a quantidade de stakeholders é muito grande.** Tem baixo custo, é fácil de aplicar, pode atingir várias pessoas, demanda menos tempo e fornece rápido feedback. No entanto, há uma tonelada de problemas, como falta de interação, perguntas pouco objetivas, difícil compreensão das perguntas, entre outros.

**Diferentemente da entrevista, essa técnica é interessante quando temos uma quantidade muito grande de pessoas para extrair as mesmas informações, que não seria viável por meio de entrevistas individuais.** As questões são dirigidas por escrito aos participantes com o objetivo de ter conhecimento sobre opiniões das mesmas questões – são autoaplicáveis, pois o próprio informante responde.

### 5. Workshop de Requisitos



**Reunião estruturada e intensiva entre analistas e usuários com o intuito de obter um conjunto de requisitos bem definidos.** Possui um facilitador neutro responsável pelas atividades de logística e promoção de momentos de descontração, como forma de dinamizar o trabalho em equipe. Permite utilizar outras técnicas em conjunto como *brainstorming* ou interpretação de papéis.

Devem fazer parte do grupo uma equipe de analistas e uma seleção/amostra dos stakeholders que melhor representam a organização e o contexto em que o sistema será utilizado, obtendo assim um conjunto de requisitos bem definidos. **Por ser realizado por convocação por dia e horário, pode ocasionar problemas por conta da presença física dos stakeholders,** mas isso tem mudado pós-pandemia.

## 6. Brainstorming (Tempestade de Ideias)

É uma abordagem de elicitação ocorrida em grupo em ambientes informais durante cerca de 15 minutos em que toda a ideia deve ser levada em consideração, sendo proibida a crítica a qualquer sugestão dada, e encorajada, inclusive, a criação de ideias que pareçam estranhas ou exóticas. **Busca-se explorar a potencialidade criativa de um grupo – um facilitador organiza e prioriza os resultados.**

Dentre suas vantagens, podemos afirmar que várias pessoas pensam melhor do que uma. **Além disso, a técnica de tempestade de ideias democratiza a participação de membros do grupo. Uma pequena desvantagem é que ela depende da disponibilidade dos integrantes.** O brainstorming muitas vezes é utilizado em conjunto com outras técnicas, tais como workshops de requisitos.

## 7. Leitura de Documentos

**A técnica de leitura de documentos é responsável por coletar informações que são geralmente mais difíceis de se obter por meio de entrevistas, questionários e observações sociais,** como – por exemplo – histórico da organização, cultura e hábitos internos, relacionamentos setoriais, informações financeiras e direcionamentos futuros. São muito utilizadas no contexto de sistemas legados.

**Estudo e reutilização de documentação de diferentes naturezas, para a identificação de requisitos a serem implementados no sistema que se está modelando, podem ser utilizados.** Uma grande variedade de documentação pode ser analisada incluindo estrutura organizacional da empresa, padrões de mercado, leis, manuais de usuário, relatório de pesquisas de mercado, glossário de termos de negócio, etc.

## 8. JAD (Joint Application Design)

Similar à técnica de workshop de requisitos e registrada pela IBM, ela busca reunir os usuários e desenvolvedores em um workshop estruturado para levantar requisitos e promover a tomada de decisões por meio de diversos tipos de dinâmicas de grupo, técnicas visuais, processos racionais e



até documentação. **É bastante interativa e promove a participação ativa dos envolvidos – inclusive dos tímidos.**

O processo consiste em três fases principais: customização, sessões e agrupamento. Na customização, o analista prepara as tarefas para as sessões como organizar os times, preparar o material, entre outros. **Na fase de sessões, o analista marca uma ou mais reuniões com os stakeholders.** No início da sessão, o engenheiro de requisitos provê uma visão genérica sobre o sistema.

A discussão com os stakeholders continua até o fim do levantamento de requisitos. **Na fase de agrupamento todos os requisitos levantados nas fases anteriores são convertidos em documentos de especificação de requisitos.** As discussões que ocorrem na fase de sessões são altamente produtivas, porque resolvem dificuldades entre as partes enquanto se dá o desenvolvimento do sistema para a empresa.

## 9. Prototipação

**Técnica de elicitação, independente de tecnologia, utilizada no estágio inicial do projeto, ajudando stakeholders a desenvolverem uma forte noção sobre a aplicação a ser implementada.** Por meio da visualização de um esboço da aplicação, podem-se identificar requisitos reais e fluxos de trabalho do sistema. São frequentemente utilizadas quando os usuários são incapazes de expressar suas necessidades.

Permite alcançar um feedback antecipado dos stakeholders e reduzir o tempo e o custo de desenvolvimento devido a detecção dos erros em uma fase inicial do projeto. **Ela fornece também alto nível de satisfação dos usuários devido a sensação de segurança ao ver algo próximo do real.** No entanto, há um alto custo de investimento em relação a outras técnicas estudadas.

## 10. Reúso de Requisitos

O reúso de requisitos trata do estudo e reutilização de especificações e glossários referentes a projetos de sistemas legados ou sistemas de mesma família ou com funcionalidades de negócio similares. **Estudos mostram que sistemas similares podem reutilizar mais de 80% de seus requisitos.** Dessa forma, eles têm chances maiores de serem compreendidos pelos *stakeholders* – o que é uma excelente vantagem.

Além disso, essa técnica economiza tempo e dinheiro! Estudos têm mostrado que sistemas similares podem reutilizar acima de 80% de seus requisitos. Pode levar também a uma reutilização adicional de outros itens em outras atividades do ciclo de vida de desenvolvimento. **Além disso, reduz riscos, visto que requisitos reutilizados têm uma chance maior de serem compreendidos pelos stakeholders.**

## 11. Histórias de Usuários



Introduzida inicialmente pela Metodologia XP (*Extreme Programming*), nada mais é do que uma história contada na linguagem do usuário final, que deve ser capaz de capturar aquilo que o usuário de fato necessita fazer para realizar seu trabalho. **Deve ser concisa o suficiente para caber em um post-it.** Um padrão seria: "Como um <papel>, eu quero <meta> de modo que <benefício>". Vamos supor que você está planejando uma viagem de férias.

Você tem em mente alguns lugares que gostaria de visitar, atividades que quer fazer e uma ideia geral de quanto tempo gostaria de ficar em cada lugar. **A história do usuário é como se fosse uma dessas atividades específicas que você quer fazer em um determinado lugar, e que contribuirá para atingir o objetivo final da viagem.** Por exemplo, se você estiver planejando uma viagem para a praia, uma história do usuário pode ser:

## "COMO UM TURISTA, EU QUERO ALUGAR UM EQUIPAMENTO DE SNORKEL PARA PODER EXPLORAR O FUNDO DO MAR E TER UMA EXPERIÊNCIA ÚNICA"

Assim como na viagem, as histórias de usuário são pequenas partes do todo que ajudam a alcançar o objetivo final do projeto de software. Elas são detalhadas o suficiente para que os desenvolvedores possam entendê-las e implementá-las, mas também são flexíveis o suficiente para permitir ajustes e mudanças à medida que o projeto avança. **Essa é uma das técnicas mais utilizadas e mais cobradas em prova (apesar de ser bem mais cobrada no contexto de XP).**

Ainda nesse contexto, temos os critérios de aceitação, isto é, condições que uma história do usuário deve satisfazer para ser considerada completa e funcionando como desejado. **Eles fornecem um conjunto de diretrizes específicas para verificar se a história atingiu seu objetivo** e servem como uma lista de verificação que guia tanto os desenvolvedores quanto os testadores para garantir que todas as funcionalidades necessárias estejam presentes e funcionando corretamente.

Eles também ajudam a esclarecer a história do usuário e a definir os limites da funcionalidade. Os critérios de aceitação devem ser claros, específicos e mensuráveis. **Eles ajudam a evitar mal-entendidos e garantem que todos tenham a mesma compreensão do que está sendo solicitado.** Dado a história de usuário apresentada anteriormente sobre o turista e sua viagem ao fundo do mar, poderíamos ter os seguintes critérios de aceitação:

- **Disponibilidade de Equipamento:** o sistema deve mostrar a disponibilidade do equipamento de snorkel para as datas selecionadas pelo usuário;
- **Seleção de Equipamento:** o usuário deve ser capaz de selecionar o tipo de equipamento de snorkel desejado (Ex: tamanho, marca, etc);
- **Informações do Aluguel:** o sistema deve fornecer detalhes sobre o custo do aluguel, incluindo qualquer taxa adicional (por exemplo, seguro, entrega);



- **Processo de Reserva:** o usuário deve ser capaz de reservar o snorkel fornecendo informações necessárias, como datas de aluguel, local de entrega e informações de pagamento;
- **Confirmação da Reserva:** após a reserva, o usuário deve receber uma confirmação por e-mail com detalhes da reserva, incluindo instruções para retirada/entrega do equipamento;
- **Cancelamento e Alterações:** o sistema deve permitir o cancelamento ou alteração da reserva até 24 horas antes da data de início do aluguel, sem penalidades;
- **Instruções de Uso:** junto com o equipamento de snorkel, o usuário deve receber instruções claras sobre como usar e cuidar do equipamento durante o aluguel;
- **Suporte ao Cliente:** o usuário deve ter acesso a um serviço de suporte ao cliente para resolver quaisquer problemas ou dúvidas relacionadas ao aluguel do equipamento.

Estes critérios de aceitação garantem que a funcionalidade de aluguel de snorkel atenda às necessidades do turista, proporcionando uma experiência de usuário clara, segura e satisfatória.

Por fim, uma história de usuário pode ser vista como a composição de três componentes principais, conhecidos como **Três C's (Cartão, Conversação e Confirmação)**. Vejamos:

<b>CARTÃO</b>	O cartão é uma breve descrição da história de usuário escrita em um formato padrão, geralmente em cartões físicos ou em uma ferramenta de gerenciamento de projetos. Ele deve ser conciso, mas suficiente para lembrar a equipe do que se trata a história. Um formato comum para escrever histórias de usuário é: "Como [tipo de usuário], eu quero [alguma funcionalidade] para [alguma razão/benefício]". Exemplo: "Como um cliente, eu quero poder adicionar produtos ao meu carrinho de compras para que eu possa comprá-los posteriormente".
<b>CONVERSAÇÃO</b>	A conversa refere-se ao diálogo contínuo entre os membros da equipe e os stakeholders (como os clientes ou usuários finais) para esclarecer os detalhes e entender melhor a história de usuário. Essa conversa é crucial para assegurar que todos tenham uma compreensão comum das necessidades e dos requisitos, permitindo que a equipe desenvolva a funcionalidade de maneira adequada. As conversas podem acontecer durante as reuniões de planejamento, sessões de refinamento do backlog, ou informalmente conforme necessário.
<b>CONFIRMAÇÃO</b>	A confirmação envolve a definição de critérios de aceitação que especificam as condições que devem ser atendidas para que a história de usuário seja considerada concluída. Esses critérios de aceitação são acordados entre a equipe e os stakeholders e servem como base para testar a funcionalidade e garantir que ela atende aos requisitos especificados. Exemplo de critérios de aceitação: "O cliente deve ser capaz de adicionar até 10 itens diferentes ao carrinho. O carrinho deve exibir corretamente o nome, quantidade e preço de cada item. O total do carrinho deve ser atualizado automaticamente quando novos itens são adicionados".

## 12. Participação Ativa de Usuários





**A técnica de participação ativa de usuários permite a incorporação dos usuários e clientes ao grupo de engenharia de requisitos.** Os usuários precisam aprender as linguagens de modelagem utilizadas para ser capaz de ler as descrições e criticá-las. Permite uma interação real entre clientes e usuários, no entanto necessita de um certo treinamento aos usuários participantes.

### 13. Encenação

É uma abordagem que implica usar uma ferramenta para ilustrar para os usuários (atores) como o sistema se ajustará à organização e também indicar como ele se comportará. **Um facilitador mostra uma encenação para o grupo e este último faz comentários.** Ajuda a restringir requisitos, estimula soluções mais criativas e a revisão em equipe, facilita o processo de entrevista, entre outros.

### 14. Interpretação de Papeis

**A técnica de interpretação de papéis é uma abordagem que atribui a cada membro do grupo um papel de interesse para o sistema.** O grupo inspecionará então como o sistema é usado. Ao longo do caminho, haverá discussões sobre quem é responsável por o quê. O Analista de Sistemas interpreta o papel do usuário, o que o ajuda a obter um discernimento real do domínio do problema.

### 15. Grupo Focal

A técnica de grupo focal trata de um grupo de discussão informal e de tamanho reduzido (geralmente até 12 pessoas), com o propósito de obter informação qualitativa em profundidade. **As pessoas são convidadas para participar da discussão sobre determinado assunto.** Possui baixo custo, resposta rápida e flexibilidade, obtendo informações de qualidade em um curto prazo.

**Ele é muito eficiente para esclarecer questões complexas no desenvolvimento de projetos.** Por outro lado, exige facilitador/moderador com experiência para conduzir o grupo. Além disso, essa técnica não garante total anonimato – que é relevante em algumas ocasiões. Por fim, ele depende da seleção criteriosa dos participantes e as informações obtidas não podem ser generalizadas.

### 16. Análise de Protocolos

**Essa técnica consiste em analisar o trabalho de determinada pessoa por meio de verbalização, estabelecendo a racionalidade utilizada na execução de tarefas.** É feita por meio da pergunta "O que você faria se...?" e, assim, possibilita eliciar fatos não facilmente observáveis e permite melhor entendimento dos fatos. Galera, funciona como um protocolo passo a passo.

### 17. Pontos de Vista (*Viewpoint-Oriented Requirements Definition – VORD*)

**A técnica de pontos de vista (também chamada de VORD) considera as perspectivas de diversas partes interessadas sobre os requisitos do sistema de software.** Ela reconhece os pontos de vista dos stakeholders e fornece um framework para se tentar descobrir conflitos nos requisitos



propostos por cada um deles. Eles podem ser de três tipos principais: interação, indiretos e de domínio.

## 18. Card Sorting

**Card Sorting é uma técnica de design centrada no usuário usada principalmente para organizar informações de maneira intuitiva e lógica, de acordo com a percepção dos usuários finais.** É uma ferramenta valiosa para entender como as pessoas categorizam e rotulam informações. Na técnica de Card Sorting, os participantes são apresentados a uma série de "cartões", cada um representando um item de conteúdo, funcionalidade ou informação.

Eles são então solicitados a organizar esses cartões em grupos que façam sentido para eles. **O Card Sorting é uma técnica relativamente simples e de baixo custo, capaz de oferecer insights valiosos e informações bem estruturadas sobre como os usuários organizam informações.** Existem diferentes formas de realizar o Card Sorting, que podem ser adaptadas às necessidades do projeto. Vejamos:

- **Card Sorting Aberto:** os participantes agrupam os cartões de acordo com suas próprias categorias e criam os nomes para esses grupos;
- **Card Sorting Fechado:** os participantes agrupam os cartões em categorias predefinidas pelo pesquisador ou designer.
- **Card Sorting Híbrido:** combina elementos de Card Sorting aberto e fechado. Os participantes podem usar categorias predefinidas ou criar novas categorias se acharem necessário.

---

Galera, é possível se aprofundar bastante em cada uma dessas técnicas. **No entanto, eu acredito que dentro do contexto de Engenharia de Requisitos, basta saber uma breve descrição.** Aprofundar-se em cada uma é inviável e pouco eficiente. Ademais, é possível aprender mais dentro do contexto em que são mais utilizadas (Ex: Histórias de Usuário, no contexto de XP; Casos de Uso, no contexto de UML; entre outros). *Fechou?* 😊



## Especificação de Requisitos

**Trata-se da atividade de traduzir as informações coletadas durante a atividade de elicitação e análise em um documento que define um conjunto de requisitos.** No entanto, essa fase se diferencia da documentação de requisitos da fase anterior – o objetivo lá era criar uma documentação preliminar dos requisitos que serviria apenas para os engenheiros de requisitos; o objetivo aqui é criar uma documentação que sirva como um contrato entre as partes.

**Dessa forma, essa documentação deve servir tanto para engenheiros de requisitos quanto para os clientes.** Esse documento contém requisitos de usuário e requisitos de sistema, logo ambas as partes podem consultá-lo de maneira sistemática para verificar o que foi levantado, a análise de viabilidade, os requisitos mais abstratos, os requisitos mais técnicos, o que foi acordado entre as partes, entre outros. *Entendido?*

Na parte de requisitos de usuário, o documento pode utilizar uma linguagem natural, com tabelas simples, diagramas ou imagens; na parte de requisitos de sistema, o documento pode utilizar modelos matemáticos formais, cenários de casos de uso, entre outras técnicas. **Galera, idealmente, requisitos de usuário e sistema devem ser claros, não-ambíguos, fáceis de entender, completos e consistentes.** *Diego, por que idealmente?*

**Porque é praticamente impossível garantir que tudo será claro, que não haverá nenhuma ambiguidade, que todos que lerem entenderão facilmente, que será bastante completo e não faltará nada, e que o documento não possui nenhuma inconsistência.** Além disso, mesmo que encontremos um cliente que concordo com tudo isso sobre o documento, podemos ter outro cliente da mesma organização que tem uma opinião totalmente diferente.

Clientes podem interpretar requisitos de maneiras diferentes. *Bacana?* Agora, um detalhe: essa não é a fase responsável por verificar se os requisitos estão claros, não-ambíguos, consistentes, fáceis de entender, etc – ela busca apenas escrever um documento de requisitos de forma clara, não-ambígua, consistente, fáceis de entender, etc. **Por outro lado, a fase responsável por efetivamente verificar esses atributos é a fase de Validação de Requisitos.**

*Vamos resumir?* **Essa é a fase que buscará escrever os requisitos em um Documento de Requisitos**<sup>1</sup>. Idealmente, requisitos de usuário e sistema devem ser claros, não-ambíguos, fáceis de entender, completos e consistentes. Na prática, isso é extremamente difícil de se atingir, na medida em que as partes interessadas interpretam os requisitos de maneiras diferentes e, frequentemente, há conflitos e inconsistências.

Deve-se descrever os requisitos de usuário em linguagem mais simples, especificando somente características externas, evitando características do projeto de sistema. Já os requisitos de sistemas

---

<sup>1</sup> Esse artefato pode ser um documento escrito (Ex: Textos), um modelo gráfico (Ex: Diagramas), um modelo matemático formal (Ex: Autômatos), cenários de casos de uso (em geral, para sistemas menores).



são extensões dos requisitos de usuário e são usados como ponto de partida para o projeto do sistema. **Eles adicionam detalhes e explicam como os requisitos do usuário devem ser fornecidos pelo sistema.**

**Por fim, essa fase gera o conjunto de requisitos que, na próxima fase, apenas será validada.** Ele apresenta uma visão do sistema e é bastante útil em diversas áreas de engenharia, descrevendo as funcionalidades de um sistema de software e suas limitações. Ademais, permite detalhar as informações de entrada e saída do sistema, de modo que se implemente uma arquitetura confiável do sistema.

**(MPU – 2010)** A especificação de requisitos permite, em determinado momento, revelar o que o sistema irá realizar no que se refere às funcionalidades, sem definir, nesse momento, como as funcionalidades serão implementadas.

**Comentários:** a especificação de requisitos busca criar o documento de requisitos e revelar apenas o que sistema realizará, mas não como ele o fará com detalhes de projeto, de tecnologias, etc (Correto).

**(TCE/PR – 2016 – Letra A)** O documento de especificação de requisitos é um documento restrito à equipe de desenvolvimento de software.

**Comentários:** é um documento que serve tanto para a equipe de desenvolvimento de software quanto para o cliente (Errado).

**(TSE – 2011 – Item IV)** Na especificação de requisitos, pode-se construir um documento que descreva o sistema em termos gerais. Esse documento apresenta uma visão do sistema e pode capturar as necessidades dos usuários.

**Comentários:** a especificação de requisitos realmente constrói um documento que descreve de forma geral o sistema por meio de uma visão sobre as necessidades dos usuários. A redação da questão peca um pouco por utilizar "termos gerais", porque pode dar a entender que se trata de um documento mais abstrato, sendo que o documento de especificação de requisitos é um documento que apresenta mais detalhes (Correto).



## Validação de Requisitos

A atividade de Validação de Requisitos é responsável por verificar os requisitos em relação ao realismo, consistência, abrangência, validade, completude, etc. Durante esse processo, erros no documento de requisitos são inevitavelmente descobertos. Devem, então, ser feitas modificações para corrigir esses problemas. **Também se busca demonstrar se os requisitos definem, de fato, o que o usuário deseja em seu sistema.**

**Este estágio é focado no cliente e está relacionado à descoberta de problemas com requisitos.** A validação de requisitos é extremamente importante porque os erros em um documento de requisitos podem levar a custos excessivos de retrabalho quando são descobertos durante o desenvolvimento ou depois que o sistema está em operação. O custo de correção de um problema ocorrido na fase de requisitos é muito maior do que aquele ocorrido na fase de projeto e codificação.

A razão disso é que uma **mudança de requisitos significa geralmente que o projeto e a implementação do sistema devem também ser mudados** e o sistema deve ser novamente testado. Não se deve subestimar os problemas de validação de requisitos. É difícil demonstrar que um conjunto de requisitos atende às necessidades do usuário. Os usuários devem imaginar o sistema em operação e avaliar sua adequação ao trabalho.

É difícil para profissionais de informática habilidosos realizarem esse tipo de análise abstrata e é ainda mais difícil para os usuários do sistema. Como resultado, raramente encontram-se todos os problemas de requisitos durante o processo de validação. **É inevitável que haja mudanças de requisitos posteriores para corrigir omissões e mal-entendidos depois da aprovação do documento de requisitos.**

Enfim, uma série de técnicas de validação de requisitos pode ser usada, tais como: Revisão de Requisitos, Prototipação e Geração de Casos de Teste.

### 1. Revisão de Requisitos (Revisão Técnica):

Requisitos são analisados sistematicamente por uma equipe de revisores. Em revisões informais, a equipe pode simplesmente ter uma conversa, envolvendo o maior número possível de representantes dos stakeholders, acerca dos requisitos produzidos. **Em revisões formais, a equipe de revisores deve confirmar junto do cliente um conjunto de critérios que todos os requisitos devem cumprir.**

**Em uma revisão formal de requisitos, a equipe de desenvolvimento deve 'conduzir' o cliente pelos requisitos de sistema, explicando as implicações de cada um dos requisitos.** A revisão ocorre sob quais critérios, Diego? A equipe de revisão deve verificar cada requisito em termos de consistência (para evitar ambiguidade), bem como verificar os requisitos como um todo em termos de completeza.



**Conflitos, contradições, erros e omissões nos requisitos devem ser apontados e registrados formalmente no relatório de revisão.** É, portanto, de responsabilidade dos usuários, do adquirente do sistema e do desenvolvedor de sistema negociar uma solução para esses problemas. Os revisores podem verificar a facilidade de verificação e compreensão, rastreabilidade e adaptabilidade dos requisitos.

Neste planejamento, devem ser preparadas checklists genéricas de revisão que não deverão incidir sobre requisitos individuais, mas sobre as relações entre requisitos, assim como as propriedades de qualidade do documento. Os seguintes atributos devem ser levados em consideração: **Compreensibilidade; Redundância; Completude; Consistência; Organização; Conformidade; e Rastreabilidade.**

ATRIBUTOS	DESCRIÇÃO
VALIDADE	Examina se as partes interessadas que contribuíram com o levantamento de requisitos aceitam a especificação final obtida.
CONSISTÊNCIA	Examina se existem conflitos entre os requisitos identificados.
COMPREENSIBILIDADE	Examina se os requisitos são compreendidos de forma inequívoca pelas partes interessadas.
COMPLETUDE	Examina se todas as funcionalidades pretendidas fazem parte da especificação do sistema.
REALISMO	Examina se, dadas as restrições do projeto (tecnológicas, financeiras e temporais), o sistema especificado é implementável.
VERIFICABILIDADE	Examina se os requisitos foram descritos de forma que seja possível verificar se foram ou não implementados.
RASTREABILIDADE	Examina se a origem de cada requisito está claramente identificada.
ADAPTABILIDADE	Examina se os requisitos podem sofrer alterações sem produzir efeitos em outros requisitos.
CONFORMIDADE COM NORMAS	Examina se a especificação obedece às normas técnicas utilizadas para o desenvolvimento do sistema.

**Ademais, podemos dizer que a Revisão Técnica se divide em Comentários, Inspeções e Walkthroughs.** No primeiro caso, os requisitos são repassados e são realizados comentários; no segundo caso, busca-se antecipar a descoberta de falhas, lendo, entendendo o documento e checando por meio de um checklist de modo que não se propaguem para o passo seguinte do processo de software.

**Por fim, os Walkthroughs são realizados através de uma execução passo a passo de um procedimento ou programa (no papel), com a finalidade de encontrar erros.** São realizadas simulações da execução por cada revisor, controlada por um testador que durante a reunião disponibiliza um conjunto de casos de teste e monitora os resultados obtidos de cada revisor. *Interessante, não?*





De acordo com Stephen R. Schach, de forma superficial, a diferença entre uma inspeção e um walkthrough é que a equipe de inspeção usa uma lista de verificação de questões levantadas para ajudá-la a encontrar as imperfeições. **No entanto, a diferença vai muito além disso! Walkthrough é um processo de duas etapas: preparação, seguida de análise do documento pela equipe.**

Inspeção é um processo de cinco etapas: visão geral, preparação, inspeção, reformulação e acompanhamento; o procedimento a ser seguido em cada etapa é formalizado. **Exemplos dessa formalização são a categorização metódica das falhas e o emprego dessas informações na inspeção dos documentos dos fluxos de trabalho seguintes bem como nas inspeções de futuros produtos.**

## 2. Prototipação:

Um modelo executável do sistema é apresentado para usuários finais e clientes. **Eles podem experimentar o modelo para verificar se atende às suas necessidades reais.** Também há desvantagens: o tempo gasto na sua implementação pode não justificar o seu uso, pode enviesar os usuários e pode ainda levar os programadores a cair na tentação de usar o protótipo para continuar o desenvolvimento do sistema.

Professor, me ajuda! *Qual a diferença entre a técnica de prototipação mencionada na fase de Elicitação de Requisitos e a técnica de prototipação mencionada agora na fase de Validação de Requisitos?* **Bem, no primeiro caso, o objetivo é descobrir, levantar, elicitar novos requisitos do sistema. No segundo caso, é validar – por meio de um protótipo – se os requisitos elicitados são realmente o que o usuário pensava.**

## 3. Geração de Casos de Teste:

**É importante destacar que os requisitos devem ser testáveis.** Se os testes dos requisitos forem criados como parte do processo de validação, eles frequentemente revelarão problemas de requisitos. Se um teste for difícil demais ou impossível de ser projetado, significa geralmente que os requisitos serão difíceis de serem implementados e devem ser reconsiderados para implementação.

**A diferença entre Verificação e Validação de Requisitos é, em geral, ignorada em prova – ambos são chamados apenas de Validação de Requisitos.** Caso cobrem, eis a diferença: a Verificação de Requisitos tem o objetivo de descobrir se os requisitos são claros, precisos, completos e consistentes, e tem por objetivo analisar se os modelos construídos estão de acordo com os requisitos definidos.

*Professor, e a validação de requisitos?* **Ela se ocupa de mostrar que os requisitos realmente definem o sistema que o cliente deseja, isto é, visa assegurar que as necessidades do cliente estão sendo atendidas por tais requisitos. Entenderam um pouco melhor?** No entanto, conforme eu disse, é comum que as provas tratem ambos simplesmente como Validação de Requisitos.



Durante o processo de validação de requisitos, devem ser realizadas verificações nos requisitos do documento de requisitos. Essas verificações incluem:

TIPO DE VERIFICAÇÃO	DESCRIÇÃO
VERIFICAÇÃO DE VALIDADE	Estudos e análises podem identificar que funções adicionais e diferentes daquelas levantadas pelos usuários são necessárias. Os sistemas têm diversos stakeholders com necessidades diferentes e qualquer conjunto de requisitos é, inevitavelmente, um compromisso.
VERIFICAÇÃO DE CONSISTÊNCIA	Os requisitos em um documento não devem ser conflitantes e contraditórios. Isso significa que não devem existir restrições ou descrições contraditórias para a mesma função do sistema, porque isso gera problemas sérios de inconsistência.
VERIFICAÇÃO DE COMPLETEZA	O documento de requisitos deve incluir requisitos que definam todas as funções e as restrições desejadas pelo usuário do sistema. Isso torna o documento de requisitos bastante verboso e pesado, no entanto é um requisito para alcançar a completeza do requisito.
VERIFICAÇÃO DE REALISMO	Usando o conhecimento da tecnologia existente, os requisitos devem ser verificados quanto a se realmente podem ser implementados. Essas verificações também devem levar em consideração o orçamento e o prazo para o desenvolvimento do sistema – caiu questão de prova recente sobre isso!
FACILIDADE DE VERIFICAÇÃO	Para reduzir o potencial de divergências entre cliente e fornecedor, os requisitos do sistema devem sempre ser escritos de modo que sejam verificáveis. Isso significa que você deve ser capaz de escrever um conjunto de testes que possa demonstrar que o sistema entregue atende a cada requisito especificado.

*O que vocês precisam memorizar sobre a validação de requisitos? Vocês devem saber que se trata de uma etapa para verificar duas coisas: se os requisitos atendem, de fato, às necessidades dos usuários; e se os requisitos são válidos, consistentes, completos, reais, abrangentes, etc. **Para isso, existem diversas técnicas que podem ser utilizadas isoladamente ou em conjunto.** Fechado? Então vamos para a nossa última fase...*



## Gerenciamento de Requisitos

Enfim, finalizamos as etapas de Engenharia de Requisitos. **No entanto, há um processo extremamente importante que envolve todas as fases estudadas: Gerenciamento de Requisitos.** Sabe-se que os requisitos estão em constante evolução. Portanto, esse é o processo responsável por compreender, acompanhar e controlar as mudanças dos requisitos de sistema, identificando inconsistências.

É necessário manter o acompanhamento dos requisitos individuais e manter as ligações entre os requisitos dependentes, **de modo que seja possível avaliar o impacto das mudanças de requisitos** (rastreabilidade). É necessário, também, estabelecer um processo formal para fazer propostas de mudança e ligá-las aos requisitos. O processo de gerenciamento de requisitos deve se iniciar assim que uma versão inicial do documento de requisitos esteja disponível.

**No entanto, o planejamento das mudanças de requisitos deve ser iniciado durante o processo de elicitação de requisitos.** A evolução de requisitos, durante o processo de engenharia de requisitos e após a entrada de um sistema em operação, é inevitável. O desenvolvimento de requisitos de software enfoca as capacidades de software, objetivos da empresa e outros sistemas da empresa.

À medida que a definição dos requisitos se desenvolve, uma compreensão maior das necessidades dos usuários é obtida. Isso **realimenta as informações do usuário que pode, então, propor uma mudança nos requisitos.** Existem vários relacionamentos entre os requisitos em si, entre os requisitos e componentes e entre requisitos e o projeto do sistema. Há também ligações entre requisitos e os motivos básicos de porque esses requisitos foram propostos.

**Quando as mudanças são propostas, deve-se rastrear seu impacto em outros requisitos e no projeto do sistema.** A rastreabilidade é a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados. Ela é frequentemente representada por meio de matrizes de rastreabilidade que relacionam os requisitos aos stakeholders, aos outros requisitos, aos módulos de projeto, aos artefatos ou subprodutos, entre outros.

Em uma matriz de rastreabilidade de requisitos, cada requisito é introduzido em uma linha e uma coluna da matriz. **As dependências entre diferentes requisitos são registradas na célula correspondente à intersecção de linha e coluna.** Existem três tipos de informações de rastreabilidade que podem ser mantidas na matriz – elas são apresentadas como pode ser visto a seguir:

- **Informações de rastreabilidade da origem:** ligam os requisitos aos stakeholders que propuseram os requisitos e aos motivos desses requisitos. Quando uma mudança é proposta, usam-se essas informações para consultar os stakeholders sobre a mudança.



- **Informações de rastreabilidade de requisitos:** ligam os requisitos dependentes dentro do documento de requisitos. Usam-se essas informações para avaliar quantos requisitos serão afetados pela mudança e a extensão das mudanças de requisitos necessárias.
- **Informações de rastreabilidade de projeto:** ligam os requisitos aos módulos de projeto, nos quais esses requisitos são implementados. Você usa essas informações para avaliar o impacto das mudanças de requisitos propostas no projeto e na implementação do sistema.

As matrizes de rastreabilidade podem ser usadas quando um pequeno número de requisitos deve ser gerenciado, **mas para sistemas de grande porte, com muitos requisitos, tornam-se muito difíceis de serem gerenciadas e sua manutenção é dispendiosa**. Para esses sistemas, deve-se captar as informações de rastreabilidade em um banco de dados de requisitos – é algo bem mais complexo.

Por falar nisso, um de nossos autores favoritos (Roger Pressman) afirma que o gerenciamento formal de requisitos é iniciado somente para grandes projetos com centenas de requisitos identificáveis. **Para projetos pequenos, essa função de engenharia de requisitos é consideravelmente menos formal e, na prática, dispensável**. *Bacana?* Por fim, vamos falar sobre rapidamente sobre tipos de rastreabilidade.

**Podemos classificar a rastreabilidade em horizontal ou vertical.** A rastreabilidade horizontal é a rastreabilidade entre diferentes versões ou variações de requisitos, ou outros artefatos, em uma particular fase do ciclo de vida; enquanto a rastreabilidade vertical é realizada entre requisitos e artefatos produzidos pelo processo de desenvolvimento ao longo do ciclo de vida do projeto. No entanto, temos mais uma classificação importante:

- **Rastreabilidade de Frente-Para (Forward-to Traceability):** rastreabilidade de origens (requisitos de clientes, requisitos no nível de sistema, etc.) para requisitos.
- **Rastreabilidade de Frente-De (Forward-from Traceability):** rastreabilidade de requisitos para especificações de projeto.
- **Rastreabilidade de Trás-para (Backward-to Traceability):** rastreabilidade de especificações de projeto para requisitos.
- **Rastreabilidade de Trás-de (Backward-from Traceability):** rastreabilidade de requisitos para suas origens (requisitos de clientes, requisitos no nível de sistema, etc.).



## RESUMO

### CLASSIFICAÇÃO QUANTO AO NÍVEL DE ABSTRAÇÃO

<b>REQUISITOS DE USUÁRIO</b>	Descrições, em linguagem natural e com diagramas, de quais serviços o sistema deve fornecer e as restrições sob as quais deve operar. São requisitos com alto nível de abstração e poucos detalhes, feitos para serem lidos por pessoas leigas – podem ser funcionais ou não funcionais.
<b>REQUISITOS DE SISTEMA</b>	Descrições detalhadas sobre as funções, operações e restrições de sistema que definem exatamente o que deve ser implementado. São requisitos com baixo nível de abstração e muitos detalhes, feitos para serem lidos por pessoas experientes – podem ser funcionais ou não funcionais.

### CLASSIFICAÇÃO QUANTO À QUALIDADE

<b>REQUISITOS NORMAIS</b>	Refletem os objetivos e metas estabelecidos para um produto ou sistema durante reuniões com o cliente. Se esses requisitos estiverem presentes, o cliente fica satisfeito. Exemplos de Requisitos Normais poderiam ser tipos de displays gráficos solicitados, funções de sistema específicas e níveis de desempenho definidos.
<b>REQUISITOS ESPERADOS</b>	Estão implícitos no produto ou sistema e podem ser tão fundamentais que o cliente não os declara explicitamente. Sua ausência será causa de grande insatisfação. Exemplos de Requisitos Esperados: facilidade na interação homem-máquina, confiabilidade e correção operacional global e facilidade na instalação do software.
<b>REQUISITOS FASCINANTES</b>	Esses recursos vão além da expectativa dos clientes e demonstram ser muito satisfatórios quando presentes. Por exemplo, o software para um novo celular vem com recursos-padrão, mas junto vem um conjunto de capacidades não esperadas. Exemplos de Requisitos Fascinantes: tecla multitoque e correio de voz visual.

### CLASSIFICAÇÃO QUANTO À EVOLUÇÃO

<b>REQUISITOS PERMANENTES</b>	Também chamados de Requisitos Estáveis, estão diretamente ligados a atividade principal da organização. São concebidos com a essência de um sistema e seu domínio da aplicação, e mudam mais lentamente que requisitos voláteis. Em geral, eles são derivados do Modelo de Domínio.
<b>REQUISITOS VOLÁTEIS</b>	Também chamados de Requisitos Instáveis, são específicos para a instanciação de um sistema em um ambiente ou um cliente particular e são mais propensos a mudança. Se modificam quando o sistema está em desenvolvimento ou em uso. Podem ser subclassificados em mutáveis, emergentes, consequentes ou de compatibilidade.

### CLASSIFICAÇÃO QUANTO À FUNCIONALIDADE

<b>REQUISITOS FUNCIONAIS</b>	São ações ou funcionalidades que o sistema deve fornecer para atingir seus objetivos. Eles dependem do tipo de software, dos usuários esperados e do tipo de sistema onde o software será implantado e fazem parte da arquitetura de um sistema. Grosso modo, pode-se dizer que eles tratam de o que o sistema deve fazer enquanto os requisitos não-funcionais tratam de como o sistema deve fazer.
------------------------------	--





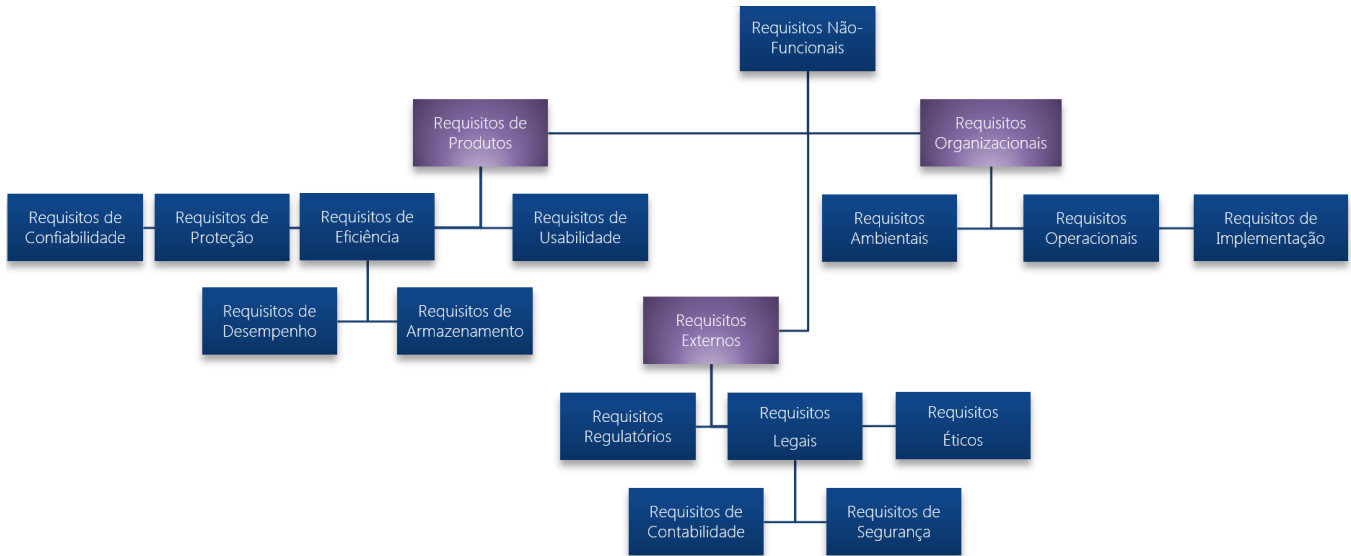
<b>REQUISITOS NÃO-FUNCIONAIS</b>	São restrições ou condições estipuladas sobre as quais o sistema deve funcionar. Não estão diretamente relacionados às funções específicas do sistema, mas às gerais – e podem incluir restrições de tempo, restrições de processo de desenvolvimento, restrições impostas por padrões, entre outras. Podem ser mais críticos que os funcionais e sempre devem ser verificáveis. Eles fazem parte da arquitetura técnica de um sistema.
<b>REQUISITOS DE DOMÍNIO</b>	são requisitos derivados do domínio da aplicação e refletem características de sua área de negócio. Eles podem ser requisitos funcionais ou não-funcionais e, caso não sejam satisfeitos, o sistema pode não ser realizável. Por exemplo, um avião que não atende aos requisitos de confiabilidade, não será certificado para voo.

<b>CLASSIFICAÇÃO QUANTO À ORIGEM</b>	
<b>REQUISITOS DE PRODUTO</b>	Especificam o comportamento do produto. Entre os exemplos, estão requisitos de desempenho quanto à rapidez com que o sistema deve operar e quanto de memória ele requer, requisitos de confiabilidade que definem a taxa aceitável de falhas, requisitos de portabilidade e requisitos de usabilidade.
<b>REQUISITOS ORGANIZACIONAIS</b>	São derivados de políticas e procedimentos da organização do cliente e do desenvolvedor. Entre os exemplos, estão padrões de processo que devem ser usados, linguagem de programação ou o método de projeto usado, e requisitos de entrega que especificam quando o produto e a sua documentação devem ser entregues.
<b>REQUISITOS EXTERNOS</b>	Abrange todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento. Entre os exemplos, estão a interoperabilidade que define como o sistema interage com outros sistemas, requisitos legais que devem ser seguidos, requisitos éticos sistema para assegurar que ele será aceito por todos.

<b>REQUISITOS NÃO-FUNCIONAIS</b>	<b>EXEMPLOS</b>
<b>REQUISITOS DE CONFIABILIDADE</b>	O sistema não deve ficar fora do ar por mais de cinco segundos durante o dia.
<b>REQUISITOS DE PROTEÇÃO</b>	O sistema não deve permitir que os usuários modifiquem senhas de acesso que eles não criaram.
<b>REQUISITOS DE DESEMPENHO</b>	O sistema deverá ser capaz de processar oitocentas requisições por segundo.
<b>REQUISITOS DE ESPAÇO</b>	Também chamado de Requisitos de Armazenamento, o sistema deverá ocupar, no máximo, 80Mb da memória interna do dispositivo.
<b>REQUISITOS DE USABILIDADE</b>	Os usuários deverão operar todas as funcionalidades do sistema após 2 horas de treino.
<b>REQUISITOS DE SEGURANÇA</b>	O sistema não deve permitir a ativação simultânea de mais de três sinais de alarme.
<b>REQUISITOS</b>	O sistema não apresentará aos usuários quaisquer dados de natureza confidencial de outrem.



<b>ÉTICOS</b>	
<b>REQUISITOS DE IMPLEMENTAÇÃO</b>	A interface de usuário deve ser implementada em HTML e não se deve utilizar Applets de Java.



FASES	DESCRIÇÃO
<b>CONCEPÇÃO</b>	Após uma necessidade de o negócio ser identificada, busca-se estabelecer um entendimento básico do problema. Trata-se da concepção inicial do software e busca entender o problema, quem são os envolvidos, a natureza da solução e iniciar o processo de comunicação entre clientes e colaboradores.
<b>LEVANTAMENTO</b>	Etapa crítica, utiliza uma abordagem organizada para descobrir o que o cliente deseja em seu sistema. Envolve intensa participação do stakeholders e faz três perguntas: Qual o objetivo do produto? Como o produto se enquadra nas necessidades do negócio? Como o produto será utilizado?
<b>ELABORAÇÃO</b>	Por vezes chamada Análise, informações obtidas do cliente durante a concepção e levantamento são expandidas e refinadas em um modelo, definindo o domínio do problema. Incluem-se modelagens de cenários de interação do usuário com o sistema e modelagens das classes envolvidas.
<b>NEGOCIAÇÃO</b>	Tem por objetivo chegar a um consenso sobre os conflitos entre clientes e usuários, por intermédio de um processo de negociação. Os requisitos são avaliados junto ao cliente e podem se combinar, excluir ou até mesmo inserir novos requisitos.
<b>ESPECIFICAÇÃO</b>	Por vezes chamada Documentação, produto final do engenheiro de requisitos, pode ser um documento escrito, um modelo gráfico, cenários de uso, protótipos, etc. Trata-se da apresentação formal dos dados obtidos até o momento de modo que possa guiar o desenvolvimento futuro do software.
<b>VALIDAÇÃO</b>	Os produtos de trabalho resultantes da engenharia de requisitos são avaliados quanto a sua qualidade por todos os envolvidos (clientes, colaboradores e usuários). Buscam-se erros de interpretação, ambiguidades e omissões.
<b>GESTÃO</b>	conjunto de atividades que auxiliam a equipe de projeto a identificar, controlar e rastrear requisitos e mudanças nos requisitos a qualquer momento . Para projetos de grande porte, é



uma fase essencial na medida em que mudanças em um requisito podem afetar diversos outros requisitos.

POR SOMMERVILLE		POR PRESSMAN
ESTUDO DE VIABILIDADE	→	CONCEPÇÃO
ELICITAÇÃO E ANÁLISE DE REQUISITOS	→	LEVANTAMENTO
	→	ELABORAÇÃO
OBTENÇÃO DE REQUISITOS	→	
CLASSIFICAÇÃO E ORGANIZAÇÃO	→	
PRIORIZAÇÃO E NEGOCIAÇÃO	→	NEGOCIAÇÃO
DOCUMENTAÇÃO DE REQUISITOS	→	
ESPECIFICAÇÃO	→	ESPECIFICAÇÃO
VALIDAÇÃO	→	VALIDAÇÃO
GESTÃO	→	GESTÃO



Onefootball  
 Alexandre Fernandes · 10 de out de 2019

## Cenas lamentáveis em novo empate do Vasco da Gama

### FASES DE ACORDO COM IAN SOMMERVILLE

**ESTÃO VIABILIZANDO ELIANA ESPECIALMENTE NO VASCO DA GAMA**

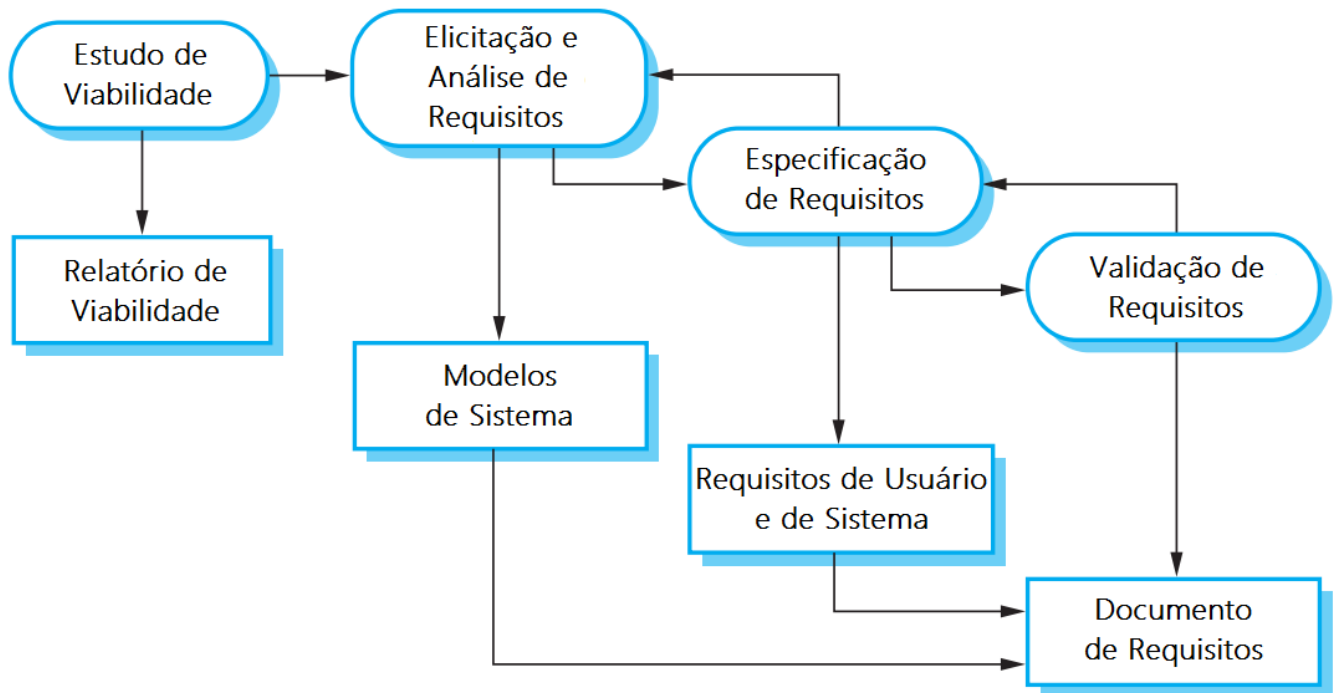
**ESTUDO DE VIABILIDADE > ELICITAÇÃO E ANÁLISE DE REQUISITOS > ESPECIFICAÇÃO > VALIDAÇÃO > GESTÃO**

### FASES DE ACORDO COM ROGER PRESSMAN

**CENAS LAMENTÁVEIS EM NOVO EMPATE DO VASCO DA GAMA**



**CONCEPÇÃO > LEVANTAMENTO > ELABORAÇÃO > NEGOCIAÇÃO > ESPECIFICAÇÃO > VALIDAÇÃO > GESTÃO**



PRINCIPAIS TÉCNICAS	DESCRIÇÃO
ENTREVISTA	Entrevistas formais ou informais com os stakeholders do sistema fazem parte da maioria dos processos de engenharia de requisitos. Nessas entrevistas, a equipe de engenharia de requisitos formula questões para os stakeholders sobre o sistema que eles usam e o sistema a ser desenvolvido. Os requisitos são derivados a partir das respostas dessas questões. As entrevistas podem ser abertas ou fechadas.
ETNOGRAFIA	Técnica de observação utilizada para compreender os requisitos organizacionais e sociais. Coloca-se o analista dentro do campo de atuação dos usuários, observando o trabalho diário anotando as tarefas reais em que os participantes estão envolvidos. Em geral, essa é uma técnica utilizada em conjunto com outras técnicas. Como ela é uma técnica de observação, isoladamente ela não é muito eficaz na elicitação.
CENÁRIOS	As pessoas geralmente consideram mais fácil relatar exemplos da vida real do que abstrair descrições. Elas podem compreender e criticar um cenário de como interagiriam com um sistema de software. Os engenheiros de requisitos podem usar as informações obtidas nessa discussão para elaborar os requisitos reais do sistema de software. Os cenários podem ser particularmente úteis para adicionar detalhes a um esboço da descrição de requisitos.
QUESTIONÁRIOS	Formulários distribuídos aos stakeholders com questões pré-definidas. Torna-se útil quando a quantidade de stakeholders é muito grande. Tem baixo custo, é fácil de aplicar, pode atingir várias pessoas, demanda menos tempo e fornece rápido feedback. No entanto, há uma tonelada de problemas, como falta de interação, perguntas pouco objetivas, difícil compreensão das perguntas, entre outros.
WORKSHOP DE REQUISITOS	Reunião estruturada e intensiva entre analistas e usuários com o intuito de obter um conjunto de requisitos bem definidos. Possui um facilitador neutro responsável pelas atividades de logística e promoção de momentos de descontração, como forma de dinamizar o trabalho em



	equipe. Permite utilizar outras técnicas em conjunto como brainstorming ou interpretação de papéis.
<b>BRAINSTORMING</b>	É uma abordagem de elicitação ocorrida em grupo em ambientes informais durante cerca de 15 minutos em que toda a ideia deve ser levada em consideração, sendo proibida a crítica a qualquer sugestão dada, e encorajada, inclusive, a criação de ideias que pareçam estranhas ou exóticas. Busca-se explorar a potencialidade criativa de um grupo – um facilitador organiza e prioriza os resultados.
<b>JOINT APPLICATION DESIGN (JAD)</b>	Similar à técnica de workshop de requisitos e registrada pela IBM, ela busca reunir os usuários e desenvolvedores em um workshop estruturado para levantar requisitos e promover a tomada de decisões por meio de diversos tipos de dinâmicas de grupo, técnicas visuais, processos racionais e até documentação. É bastante interativa e promove a participação ativa dos envolvidos – inclusive dos tímidos.
<b>PROTOTIPAÇÃO</b>	Técnica de elicitação, independente de tecnologia, utilizada no estágio inicial do projeto, ajudando stakeholders a desenvolverem uma forte noção sobre a aplicação a ser implementada. Por meio da visualização de um esboço da aplicação, podem-se identificar requisitos reais e fluxos de trabalho do sistema. São frequentemente utilizadas quando os usuários são incapazes de expressar suas necessidades.
<b>REÚSO DE REQUISITOS</b>	O reúso de requisitos trata do estudo e reutilização de especificações e glossários referentes a projetos de sistemas legados ou sistemas de mesma família ou com funcionalidades de negócio similares. Estudos mostram que sistemas similares podem reutilizar mais de 80% de seus requisitos. Dessa forma, eles têm chances maiores de serem compreendidos pelos stakeholders – o que é uma excelente vantagem.
<b>HISTÓRIAS DE USUÁRIO</b>	Introduzida inicialmente pela Metodologia XP (Extreme Programming), nada mais é do que uma história contada na linguagem do usuário final, que deve ser capaz de capturar aquilo que o usuário de fato necessita fazer para realizar seu trabalho. Deve ser concisa o suficiente para caber em um post-it. Um padrão seria: "Como um <papel>, eu quero <meta> de modo que <benefício>".

 **PARA MAIS DICAS:** [WWW.INSTAGRAM.COM/PROFESSORDIEGOCARVALHO](https://www.instagram.com/professordiegovalho)





## QUESTÕES COMENTADAS – CESPE

1. (CESPE / CAU-BR – 2024) Em engenharia de requisitos, o modelo conceitual é construído na fase de projeto do sistema de informação.

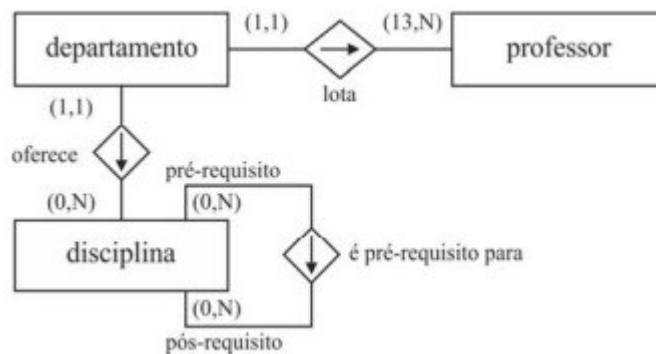
### Comentários:

Em engenharia de requisitos, o modelo conceitual é construído durante a fase de levantamento e análise de requisitos, não na fase de projeto do sistema de informação. O modelo conceitual serve para capturar e representar graficamente os requisitos do sistema, facilitando o entendimento e a comunicação entre stakeholders e desenvolvedores antes de se avançar para a fase de projeto e implementação.

**Gabarito:** Errado

2. (CESPE / CAU-BR – 2024) No contexto da engenharia de requisitos e de metodologias de desenvolvimento de software, julgue o próximo item.

No diagrama entidade e relacionamento a seguir, verifica-se um exemplo clássico de autorrelacionamento para a entidade Disciplina.



### Comentários:

No diagrama entidade e relacionamento apresentado, a entidade "Disciplina" se relaciona consigo mesma através das relações "pré-requisito" e "pós-requisito". Este é um exemplo clássico de autorrelacionamento, onde uma entidade possui relacionamentos com ela mesma, permitindo modelar hierarquias ou dependências entre os mesmos tipos de objetos. No caso, uma disciplina pode ser pré-requisito ou pós-requisito de outra disciplina.

**Gabarito:** Correto



3. (CESPE / MPE-GO – 2024) Em levantamento de requisitos baseado em cenários, um ator de casos de uso equivale tipicamente a um usuário que exerça papéis diferentes na operação do sistema a ser construído.

### Comentários:

Em levantamento de requisitos baseado em cenários, um ator de casos de uso representa uma entidade externa que interage com o sistema, podendo ser um usuário, outro sistema ou um hardware. Não se refere necessariamente a um usuário que exerce diferentes papéis, mas sim a qualquer entidade que tem uma interação significativa com o sistema para alcançar um objetivo específico.

**Gabarito:** Errado

4. (CESPE / LNA – 2024) A figura precedente ilustra um processo de elicitação e análise de requisitos. Em relação a esse processo, a técnica casos de uso



a) poderia ser utilizada na Priorização e negociação de requisitos, mas somente se o sistema for orientado a objetos, pois, nesse caso, haveria a especificação de cada caso de uso.

b) seria uma atividade do processo Descoberta de requisitos.

c) não estaria relacionada ao referido processo, pois, como o processo é um diagrama da UML, a técnica seria utilizada somente ao final do processo para a documentação do projeto.

d) está mais relacionada à Especificação de requisitos, uma vez que nela podem ser detalhadas as iterações do sistema.

e) seria melhor classificada, por envolver a descrição de quem são os atores do sistema, como uma atividade da Priorização e negociação de requisitos, uma vez que envolve a definição de papéis.



### Comentários:

- (a) Errado. A técnica de casos de uso não é limitada a sistemas orientados a objetos, e sua principal função não é a priorização, mas sim a especificação de requisitos;
- (b) Correto. A descoberta de requisitos inclui técnicas como casos de uso, que ajudam a identificar e entender as funcionalidades necessárias do sistema;
- (c) Errado. Casos de uso são utilizados durante o processo de elicitação e análise de requisitos, e não apenas na documentação final do projeto;
- (d) Errado. Casos de uso são mais utilizados na fase de Descoberta de requisitos para coletar as funcionalidades desejadas e entender o comportamento do sistema;
- (e) Errado. A definição de papéis e atores é parte da identificação de requisitos e não da priorização e negociação, que foca em definir quais requisitos devem ser implementados primeiro.

**Gabarito:** Letra B

- 
5. (CESPE / ANVISA – 2024) A entrevista com o usuário, a condição de uma sessão de brainstorming e o estudo de sistemas semelhantes são exemplos de técnicas para obtenção de requisitos.

### Comentários:

Entrevistas com usuários, sessões de brainstorming e o estudo de sistemas semelhantes são técnicas amplamente utilizadas para a obtenção de requisitos. Essas técnicas ajudam a compreender as necessidades dos usuários, gerar ideias inovadoras e identificar funcionalidades desejáveis com base em sistemas existentes. Essas abordagens são essenciais no processo de levantamento de requisitos, garantindo que o sistema a ser desenvolvido atenda às expectativas e necessidades do usuário final.

**Gabarito:** Correto

- 
6. (CESPE / MPO – 2024) Os requisitos do sistema devem descrever apenas o comportamento externo deste, logo não abrangem a forma como o sistema deve ser projetado ou implementado.

### Comentários:

De acordo com Sommerville: *"Em condições ideais, os requisitos de sistema devem descrever apenas o comportamento externo do sistema e suas restrições operacionais. Eles não devem se preocupar com o modo que o sistema deve ser projetado ou implementado. No entanto, no nível de detalhe exigido*



*para especificar completamente um sistema de software complexo, não é possível nem desejável excluir todas as informações de projeto (design)".*

Os requisitos do sistema, especificamente os requisitos funcionais, focam no comportamento externo do sistema, definindo o que ele deve fazer. Eles não detalham como o sistema deve ser projetado ou implementado, pois essas são responsabilidades da fase de design e implementação, que são especificadas em requisitos não funcionais e documentos de arquitetura. No entanto, a banca considerou que o item permite interpretação ambígua.

**Gabarito:** Anulada

---

**7. (CESPE / MPO – 2024)** No escopo da modelagem de requisitos, que visa criar várias representações do que o cliente deseja, os modelos baseados em cenários representam os requisitos de software sob o ponto de vista do usuário; nesse modelo, o caso de uso descreve a interação entre um ator e o software.

#### **Comentários:**

Na modelagem de requisitos, os modelos baseados em cenários são utilizados para representar os requisitos do software do ponto de vista do usuário. O caso de uso é uma ferramenta comum nesse contexto, descrevendo a interação entre um ator (usuário ou sistema) e o software, detalhando como o sistema deve responder a um evento ou ação específica.

**Gabarito:** Correto

---

**8. (CESPE / MPO – 2024)** Os requisitos não funcionais, mesmo não tendo relação direta com os serviços específicos oferecidos pelo sistema a seus usuários, podem estar relacionados, por exemplo, à confiabilidade e ao tempo de resposta do sistema.

#### **Comentários:**

Os requisitos não funcionais não descrevem serviços específicos que o sistema oferece, mas são cruciais para definir aspectos de qualidade e operação do sistema, como confiabilidade, desempenho, tempo de resposta, segurança e usabilidade. Eles garantem que o sistema funcione de forma eficaz e eficiente sob diversas condições.

**Gabarito:** Correto

---

**9. (CESPE / MPO – 2024)** Descrever aquilo que o sistema não deve fazer pode ser considerado, em alguns casos, como requisito funcional.

#### **Comentários:**



A afirmação está correta. Requisitos funcionais descrevem as funções e comportamentos específicos do sistema. Em alguns casos, é importante explicitar o que o sistema não deve fazer para evitar comportamentos indesejados ou ambiguidades. Esses requisitos negativos ajudam a esclarecer os limites e restrições das funcionalidades do sistema.

**Gabarito:** Correto

---

**10. (CESPE / MPO – 2024)** Por meio da técnica de elicitação de requisitos, que visa permitir descobertas de requisitos funcionais, o analista de requisitos realiza imersão no ambiente de trabalho no qual o sistema será usado e, por meio de observações, extrai os requisitos.

**Comentários:**

A técnica de elicitação de requisitos que envolve a imersão do analista no ambiente de trabalho onde o sistema será utilizado é a etnografia, que é uma técnica específica de elicitação de requisitos. Como gerou essa ambiguidade entre gênero e espécie de técnicas de elicitação de requisitos, a questão foi anulada pela banca examinadora.

**Gabarito:** Anulada

---

**11. (CESPE / MPO – 2024)** Os processos de engenharia de requisitos podem incluir quatro atividades de alto nível: validação, especificação, estudo de viabilidade e elicitação, e, nessa última etapa, descobrem-se os requisitos.

**Comentários:**

Na engenharia de requisitos, as atividades de alto nível incluem: (1) Elicitação de requisitos: Descoberta e levantamento dos requisitos junto aos stakeholders; (2) Especificação de requisitos: Documentação detalhada dos requisitos coletados; (3) Validação de requisitos: Verificação da precisão, completude e viabilidade dos requisitos; (4) Estudo de viabilidade: Avaliação se os requisitos são tecnicamente e economicamente viáveis. A elicitação é a etapa onde os requisitos são inicialmente descobertos e entendidos.

**Gabarito:** Correto

---

**12. (CESPE / FINEP – 2024)** Em engenharia de software, a busca de conformidade aos requisitos funcionais explicitamente declarados, por meio de padrões de desenvolvimento documentados para atender as características implícitas esperadas do software, é objeto da:

- a) viabilidade.
- b) governança.
- c) usabilidade.
- d) qualidade.
- e) manutenibilidade.





### Comentários:

- (a) Errado. Viabilidade geralmente se refere à análise de quão factível é um projeto ou uma proposta, considerando aspectos como custos, tempo, recursos disponíveis e impacto, não diretamente relacionado à conformidade com requisitos e padrões;
- (b) Errado. Governança em engenharia de software envolve a definição de políticas e a continuidade da gestão de projetos de TI, mas não especificamente a conformidade aos requisitos funcionais e padrões documentados do desenvolvimento de software;
- (c) Errado. Usabilidade é uma característica de qualidade do software que define o quão fácil é para o usuário final operar, entender e usar o software, não abrangendo diretamente a conformidade com requisitos funcionais documentados;
- (d) Correto. Qualidade em engenharia de software é justamente sobre atender tanto aos requisitos funcionais explícitos quanto aos padrões de desenvolvimento documentados e às expectativas implícitas de desempenho e comportamento do software. A qualidade busca garantir que o software seja confiável, eficiente e adequado ao uso pretendido;
- (e) Errado. Manutenibilidade é uma métrica de quão fácil é manter, corrigir ou adaptar um software durante sua vida útil. Embora importante, não abrange especificamente a conformidade com requisitos e padrões como descrito na questão.

**Gabarito:** Letra D

---

**13. (CESPE / MPO – 2024)** Na criação das histórias de usuários, o modelo dos 3C refere-se a: cartões, conversas e confirmações.

### Comentários:

O modelo dos 3C (Cartões, Conversas e Confirmações) é uma abordagem para a criação de histórias de usuários em metodologias ágeis. "Cartões" representam os artefatos físicos ou digitais onde as histórias são escritas. "Conversas" são as discussões colaborativas que detalham e refinam os requisitos. "Confirmações" referem-se aos critérios de aceitação que validam a completude e a adequação da história ao que foi solicitado.

**Gabarito:** Correto

---

**14. (CESPE / MPO – 2024)** As *story points* são a métrica para planejar o esforço na execução de projetos de software utilizando a análise por ponto de função.

### Comentários:



As story points são uma métrica utilizada em metodologias ágeis para estimar o esforço relativo necessário para implementar uma história de usuário, considerando fatores como complexidade, risco e volume de trabalho. Já a análise por ponto de função é uma técnica diferente que mede o tamanho funcional do software em termos de funcionalidades entregues ao usuário, não utilizando story points como unidade de medida.

**Gabarito:** Errado

**15. (CESPE / TST – 2024)** No processo de desenvolvimento de software ágil, uma narrativa que relata, sob a perspectiva do usuário final, como o software irá agregar valor ao negócio é denominada:

- a) sprint.
- b) iniciativa.
- c) história de usuário.
- d) requisito não funcional.
- e) épico.

**Comentários:**

- (a) Errado. Uma sprint é um período de tempo, geralmente de 1 a 4 semanas, durante o qual uma equipe ágil trabalha para completar um conjunto de tarefas;
- (b) Errado. Uma iniciativa é um grande esforço ou projeto que pode envolver várias equipes e sprints para alcançar um objetivo específico;
- (c) Correto. Uma história de usuário é uma descrição simples de uma funcionalidade do ponto de vista do usuário final, focando em como o software agregará valor ao negócio;
- (d) Errado. Um requisito não funcional descreve critérios que podem ser usados para julgar a operação de um sistema, mas não relata como o software agrega valor do ponto de vista do usuário;
- (e) Errado. Um épico é uma grande história de usuário que pode ser dividida em várias histórias menores.

**Gabarito:** Letra C

**16. (CESPE / CAU-BR – 2024)** Ao se especificar a nova versão de um software, devem ser mapeados os requisitos não funcionais, que definem o que o sistema deve fazer para atender as expectativas do cliente.

**Comentários:**



Os requisitos não funcionais não definem o que o sistema deve fazer, mas sim como o sistema deve se comportar. Eles descrevem atributos de qualidade do sistema, como desempenho, segurança, usabilidade, confiabilidade e escalabilidade, e são essenciais para atender às expectativas do cliente em termos de experiência e qualidade. Os requisitos funcionais, por outro lado, especificam as funcionalidades que o sistema deve oferecer.

**Gabarito:** Errado

---

**17. (CESPE / CNPq – 2024)** Após a definição dos objetivos gerais, o protótipo, nessa modelagem, atua como um mecanismo para identificar os requisitos do software.

#### **Comentários:**

Em metodologias que utilizam prototipagem, o protótipo serve como um mecanismo para identificar e refinar os requisitos do software. Após a definição dos objetivos gerais, o protótipo permite que usuários e stakeholders interajam com uma versão inicial do sistema, ajudando a esclarecer requisitos, detectar problemas e ajustar funcionalidades antes do desenvolvimento completo.

**Gabarito:** Correto

---

**18. (CESPE / CNPq – 2024)** Um caso de uso, independentemente da sua forma, representa o software ou o sistema do ponto de vista do analista.

#### **Comentários:**

Um caso de uso representa o software ou sistema do ponto de vista do usuário ou do ator, não do analista. Ele descreve como o usuário interage com o sistema para alcançar um objetivo específico, focando nas funcionalidades que o sistema deve oferecer para atender às necessidades dos usuários.

**Gabarito:** Errado

---

**19. (CESPE / DATAPREV – 2023)** A linguagem natural estruturada é uma forma de escrever os requisitos de sistema na qual a liberdade do elaborador de requisitos é limitada e todos os requisitos são redigidos de maneira padronizada.

#### **Comentários:**

A linguagem natural estruturada é uma forma de escrever os requisitos de sistema em que a liberdade do elaborador é limitada, e todos os requisitos são redigidos de maneira padronizada. Ela utiliza regras e templates predefinidos para garantir consistência e clareza, reduzindo ambiguidades comuns em linguagem natural livre. Isso facilita a comunicação entre partes interessadas e melhora a precisão na especificação dos requisitos.



**Gabarito:** Correto

---

**20. (CESPE / SEFIN de Fortaleza-CE – 2023)** Um critério de aceitação, depois de testado, significa que um requisito foi implementado corretamente.

**Comentários:**

Um critério de aceitação é uma condição ou conjunto de condições que o sistema deve atender para que um requisito seja considerado implementado corretamente. Após a realização dos testes, se o critério de aceitação for atendido, isso indica que o requisito foi implementado conforme especificado e funciona como esperado, satisfazendo as necessidades dos usuários ou das partes interessadas.

**Gabarito:** Correto

---

**21. (CESPE / SEFIN de Fortaleza-CE – 2023)** As histórias de usuários podem ser divididas em partes menores chamadas de épicas.

**Comentários:**

Histórias de usuários são descrições breves e específicas de funcionalidades desejadas do ponto de vista do usuário. Quando uma história de usuário é grande ou complexa demais para ser implementada de uma vez, ela pode ser dividida em partes menores. Já os épicas são grandes histórias que representam grandes funcionalidades ou conjuntos de funcionalidades, que, por sua vez, são divididos em várias histórias de usuários menores e mais gerenciáveis. Portanto, é o épico que pode ser dividido em histórias de usuários, e não o contrário.

**Gabarito:** Errado

---

**22. (CESPE / SEFIN de Fortaleza-CE – 2023)** A seguinte frase representa a sintaxe de uma história de usuário: como [persona], eu [quero], [para que].

**Comentários:**

A sintaxe de uma história de usuário é geralmente estruturada dessa forma: "Como [persona], eu [quero] [ação/desejo], para que [benefício/razão]". Esse formato ajuda a capturar a necessidade do usuário, o que ele deseja alcançar, e o valor ou objetivo dessa ação, facilitando a compreensão e o desenvolvimento de funcionalidades centradas no usuário.

**Gabarito:** Correto

---

**23. (CESPE / SEFIN de Fortaleza-CE – 2023)** Um requisito funcional pode ser expresso em termos de segurança e desempenho de um software.



### Comentários:

Requisitos funcionais descrevem as funcionalidades específicas que o sistema deve realizar, como cálculos, manipulação de dados e interações com o usuário. Já requisitos relacionados à segurança e desempenho são exemplos de requisitos não funcionais, que especificam critérios de qualidade, como tempo de resposta, capacidade de processamento, proteção contra acessos não autorizados, entre outros.

**Gabarito:** Errado

---

**24. (CESPE / SEFIN de Fortaleza-CE – 2023)** Requisito é uma condição necessária que o sistema deverá realizar para atender a uma necessidade do usuário.

### Comentários:

Requisito é uma condição ou capacidade que um sistema deve satisfazer para atender às necessidades dos usuários ou de outras partes interessadas. Ele descreve funcionalidades, características ou restrições que o sistema deve ter para ser considerado bem-sucedido. Os requisitos podem ser funcionais (relacionados às funções que o sistema deve executar) ou não funcionais (relacionados ao desempenho, segurança, etc.).

▪ **Gabarito:** Correto

---

**25. (CESPE / FUB – 2023)** Um protótipo pode demonstrar conceitos, validar opções de projeto e, ainda, ajudar na elicitação e validação de requisitos de um sistema.

### Comentários:

Um protótipo serve como uma representação inicial do sistema, que permite testar ideias, explorar soluções de design e obter feedback de stakeholders. Ele facilita a comunicação entre usuários e desenvolvedores, auxiliando na elicitação e validação de requisitos ao tornar visíveis e compreensíveis as funcionalidades desejadas, identificando necessidades e melhorias antes do desenvolvimento completo do sistema.

**Gabarito:** Correto

---

**26. (CESPE / FUB – 2023)** Etnografia é uma técnica de elicitação de requisitos que pode ser usada para compreender processos operacionais por meio da imersão no ambiente de trabalho em que o sistema será usado, porém é ineficaz em descobrir requisitos derivados da cooperação e do conhecimento das atividades das pessoas.

### Comentários:





A etnografia é eficaz para compreender não apenas processos operacionais, mas também para descobrir requisitos derivados da cooperação e do conhecimento das atividades das pessoas. Ao imergir no ambiente de trabalho, o analista pode observar interações e atividades que revelam necessidades e práticas colaborativas importantes para o desenvolvimento do sistema.

**Gabarito:** Errado

**27. (CESPE / MPE-RO – 2023)** Em engenharia de requisitos, os usuários respondem quais são os objetivos do software na fase de:

- a) levantamento.
- b) concepção.
- c) negociação.
- d) elaboração.
- e) especificação.

**Comentários:**

(a) Correto. A fase de levantamento (ou elicitación) de requisitos é onde os analistas de requisitos coletam informações dos usuários e stakeholders sobre o que o software deve realizar. Durante essa fase, os usuários respondem sobre os objetivos e funcionalidades que o software deve ter, fornecendo uma base para as etapas seguintes.

(b) Errado. A concepção geralmente se refere à fase em que as ideias iniciais e os conceitos gerais para o software são desenvolvidos, mas não é focada diretamente na obtenção dos objetivos específicos dos usuários.

(c) Errado. A negociação envolve discutir e priorizar os requisitos com base nas limitações do projeto, como recursos e prazos. Os objetivos do software já devem ter sido identificados antes dessa fase.

(d) Errado. A elaboração é uma fase que pode envolver a criação de modelos ou protótipos para entender melhor os requisitos, mas a identificação dos objetivos ocorre antes, no levantamento.

(e) Errado. A especificação é a fase em que os requisitos coletados são documentados de forma clara e precisa. Antes de especificar, é necessário já ter os objetivos definidos a partir do levantamento.

**Gabarito:** Letra A

**28. (CESPE / DATAPREV – 2023)** Considerados uma técnica de elicitación de requisitos, os casos de uso identificam os atores envolvidos, assim como interações individuais entre o sistema e seus usuários ou, ainda, com outros sistemas.



### Comentários:

Os casos de uso são uma técnica de elicitação de requisitos que ajudam a identificar os atores envolvidos e as interações entre o sistema e seus usuários ou outros sistemas. Eles descrevem cenários de uso, detalhando como os usuários interagem com o sistema para alcançar objetivos específicos, e são fundamentais para entender as funcionalidades e comportamentos esperados de um sistema.

**Gabarito:** Correto

**29. (CESPE / DATAPREV – 2023)** Considerando-se que o processo de engenharia de requisitos seja uma espiral iterativa com três estágios — elicitação, especificação e validação de requisitos —, é correto afirmar que, nesse contexto, a atividade de prototipação é mais bem relacionada à elicitação do que à validação de requisitos.

### Comentários:

No processo de engenharia de requisitos, a atividade de prototipação é mais relacionada à elicitação de requisitos. A prototipação ajuda a explorar e entender melhor as necessidades dos usuários, permitindo o refinamento dos requisitos iniciais com base no feedback. Embora também possa ser usada para validação, seu papel principal é facilitar a comunicação e o entendimento durante a elicitação, permitindo que os stakeholders visualizem e ajustem os requisitos antes que eles sejam formalmente especificados e validados.

**Gabarito:** Correto

**30. (CESPE / DATAPREV – 2023)** Os requisitos de usuário de um sistema devem descrever os requisitos funcionais, de modo que eles sejam compreensíveis pelos usuários do sistema que não possuem conhecimento técnico detalhado, abstendo-se de inserir requisitos não funcionais.

### Comentários:

Os requisitos de usuário de um sistema devem descrever os requisitos funcionais e não funcionais de forma compreensível para usuários que não possuem conhecimento técnico detalhado. Os requisitos funcionais descrevem o que o sistema deve fazer, enquanto os requisitos não funcionais especificam restrições de desempenho, segurança, usabilidade, entre outros. Ambos são importantes para garantir que o sistema atenda às necessidades dos usuários e às expectativas de qualidade.

**Gabarito:** Errado

**31. (CESPE / TBG – 2023)** Requisito funcional especifica o que o software deve fazer, enquanto requisito não funcional especifica como o sistema deve se comportar.



### Comentários:

Requisitos funcionais especificam as funcionalidades e comportamentos que o software deve possuir, como tarefas, serviços e funções que ele deve executar. Já os requisitos não funcionais descrevem como o sistema deve se comportar e os atributos de qualidade que ele deve possuir, como desempenho, segurança, usabilidade e confiabilidade. Ambos são essenciais para definir as expectativas e assegurar a qualidade do sistema.

**Gabarito:** Correto

---

**32. (CESPE / TC-DF – 2023)** A rastreabilidade horizontal refere-se à capacidade de seguir um requisito ou uma funcionalidade de um software desde a sua origem até seus desdobramentos e implementações subsequentes.

### Comentários:

A rastreabilidade horizontal se refere ao acompanhamento das relações entre diferentes artefatos de um mesmo estágio do ciclo de vida do software, como requisitos que se relacionam entre si. Já a rastreabilidade vertical é a capacidade de seguir um requisito ou funcionalidade desde a sua origem até seus desdobramentos e implementações subsequentes, cobrindo diferentes fases do desenvolvimento.

**Gabarito:** Errado

---

**33. (CESPE / BANRISUL – 2022)** Requisitos não funcionais de um sistema descrevem seu objetivo e dependem do tipo de software a ser desenvolvido, dos usuários esperados para o software e da abordagem geral adotada pela organização ao escrever os requisitos.

### Comentários:

Requisitos Não Funcionais (RNF) são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo software a seus usuários. A descrição da questão trata dos Requisitos Funcionais (RF).

**Gabarito:** Errado

---

**34. (CESPE / BANRISUL – 2022)** Requisitos organizacionais são requisitos de sistema amplos, derivados das políticas e dos procedimentos nas organizações do cliente e do desenvolvedor, cujas funções incluem definir como o sistema será utilizado e especificar a linguagem de programação.

### Comentários:



Requisitos Organizacionais são derivados de políticas e procedimentos da organização do cliente e do desenvolvedor. Entre os exemplos, estão padrões de processo que devem ser usados, linguagem de programação ou o método de projeto usado, e requisitos de entrega que especificam quando o produto e a sua documentação devem ser entregues.

**Gabarito:** Correto

**35. (CESPE / BANRISUL – 2022)** Os requisitos do sistema devem descrever os comportamentos interno e externo do sistema, devendo-se preocupar com a forma como ele deve ser projetado ou implementado.

#### Comentários:

Os requisitos do sistema devem descrever **apenas o comportamento externo** do sistema e suas restrições operacionais. Eles não devem se preocupar com a forma como o sistema deve ser projetado ou implementado.

**Gabarito:** Errado

**36. (CESPE / BANRISUL – 2022)** A especificação de requisitos é frequentemente composta de vários tipos de documentos e não raro abrange: visão geral; glossário; modelos do sistema; lista de requisitos funcionais e lista de requisitos não funcionais; especificação detalhada de requisitos.

#### Comentários:

A Especificação de Requisitos é um documento importante que descreve os requisitos específicos de um projeto. Esta especificação inclui vários documentos para garantir que todas as partes interessadas estejam cientes do que se espera do projeto. Estes documentos incluem uma visão geral, glossário, modelos do sistema, lista de requisitos funcionais e não funcionais, e especificação detalhada de requisitos.

**Gabarito:** Correto

**37. (CESPE / BANRISUL – 2022)** O objetivo principal da especificação é documentar todas as necessidades dos clientes e obter um aceite quanto às entregas de produto propostas.

#### Comentários:

A especificação tem como objetivo fornecer aos clientes um documento detalhado que descreva todas as suas necessidades e requisitos, além de obter um acordo entre as partes envolvidas quanto às entregas esperadas. Esta documentação pode ajudar a evitar erros e mal-entendidos entre as partes, garantindo que os requisitos do cliente sejam atendidos.



**Gabarito:** Correto

**38.(CESPE / BANRISUL – 2022)** Na execução da técnica de apprenticing (aprendizado), o engenheiro de requisitos deve questionar procedimentos operacionais complexos e pouco claros do domínio do sistema que os stakeholders desejam preservar.

**Comentários:**

O engenheiro de requisitos deve entender como os stakeholders gostariam de preservar os procedimentos operacionais complexos e pouco claros do sistema. A técnica de apprenticing permite que o engenheiro de requisitos questione os stakeholders sobre o que eles querem que seja preservado e sobre como o sistema deve ser construído para atender a essas necessidades. Ao questionar os stakeholders, o engenheiro de requisitos pode ajudá-los a compreender melhor como os procedimentos operacionais complexos e pouco claros devem ser implementados e gerenciados pelo sistema.

**Gabarito:** Correto

**39.(CESPE / BANRISUL – 2022)** Em situações em que alguma das partes interessadas não consiga expressar de forma oral as suas necessidades com clareza, recomenda-se o emprego da técnica da etnografia para o levantamento de requisitos.

**Comentários:**

A etnografia é uma abordagem qualitativa de pesquisa que permite o aprofundamento de conhecimento sobre o contexto de determinada situação. Ela permite a compreensão das necessidades das partes interessadas, mesmo que elas não consigam expressá-las de forma verbal. Ao seguir essa abordagem, é possível identificar os interesses, motivações e comportamentos que estão por trás do problema, permitindo assim a construção de soluções mais adequadas ao contexto.

**Gabarito:** Correto

**40.(CESPE / BANRISUL – 2022)** O levantamento de requisitos com casos de uso é muito eficaz para a elicitacão de requisitos não funcionais.

**Comentários:**

O levantamento de requisitos com casos de uso é eficaz para a elicitacão de requisitos funcionais, pois fornece detalhes sobre como um sistema deve se comportar. Os requisitos não funcionais, como aspectos de desempenho e segurança, são melhor identificados por meio de técnicas de análise de requisitos como entrevistas, questionários, estudos de caso, brainstorming, etc.





**Gabarito:** Errado

**41. (CESPE / BANRISUL – 2022)** A analogia é uma técnica pouco recomendada quando é necessário identificar requisitos novos, inovadores ou atrativos, em um ambiente cujo objetivo é encontrar soluções criativas.

**Comentários:**

A analogia pode ser uma ferramenta extremamente útil para identificar novos requisitos, inovadores ou atrativos. Ela é especialmente útil quando se trata de encontrar soluções criativas, pois permite que os usuários comparem elementos já existentes com outros que podem ser usados para criar algo novo. Ela também pode ajudar os desenvolvedores a visualizar o problema de forma diferente, o que pode apontar para soluções que eles não teriam encontrado de outra forma.

**Gabarito:** Errado

**42. (CESPE / BANRISUL – 2022)** A arqueologia é uma técnica apropriada quando se busca preservar todas as funcionalidades de um sistema legado em um novo sistema que reutilize as soluções e experiências existentes.

**Comentários:**

A arqueologia é uma ótima ferramenta para modernizar sistemas legados, pois ajuda a preservar o conhecimento e as funcionalidades existentes. Ela ajuda a extrair informações úteis sobre o sistema legado, permitindo que os desenvolvedores entendam melhor como ele está estruturado e como ele pode ser melhorado. Além disso, ela ajuda a preservar as soluções e experiências existentes, ajudando a evitar a necessidade de se reinventar a roda durante o processo de modernização.

**Gabarito:** Correto

**43. (CESPE / MPC-SC – 2022)** A etnografia é o processo de elicitación por meio do qual o analista de requisitos realiza uma imersão no ambiente de trabalho em que o sistema será utilizado para tornar compreensíveis os processos operacionais e auxiliar na extração dos requisitos de apoio de tais processos.

**Comentários:**

A etnografia é uma técnica de observação que pode ser utilizada para entender os processos operacionais e para ajudar a derivar os requisitos do software que apoia esses processos. Um analista deve ficar imerso no ambiente de trabalho em que o sistema será utilizado com o objetivo de observar o dia a dia e tomar nota das tarefas reais nas quais os participantes estão envolvidos. A vantagem da etnografia é que ela ajuda a descobrir requisitos implícitos do sistema, os quais refletem o verdadeiro modo de trabalho das pessoas, em vez dos processos formais definidos pela organização. Logo, a questão está correta, mas foi anulada porque o conteúdo extrapolava o edital.



**Gabarito:** Anulado

**44. (CESPE / BNB – 2022)** Para capturar os requisitos da interface de um sistema, os protótipos podem ser desenhados como mockups, mesmo que estes não permitam interações do usuário com a execução das funcionalidades.

**Comentários:**

Perfeito! Mockups são protótipos estáticos que fornecem uma visão geral da interface de usuário e ajudam os desenvolvedores a entender melhor os requisitos de design. Embora os mockups não permitam interações do usuário, eles são úteis para garantir que o design seja consistente e fácil de usar. Os mockups podem ajudar os designers a entender como os elementos de interface devem se conectar para criar uma experiência de usuário agradável. Eles podem também ajudar os desenvolvedores a construir a interface de usuário corretamente e fornecem aos desenvolvedores uma visão geral da aparência do produto final, possibilitando que eles criem os recursos corretos e que sejam adequados para a interface de usuário.

**Gabarito:** Correto

**45. (CESPE / BNB – 2022)** Um dos critérios de boa qualidade para uma história de usuário é o denominado critério pequeno, ou seja, aquele cujo desenvolvimento da história deve representar um trabalho desenvolvido dentro de um limite de tempo de duração específica.

**Comentários:**

Um princípio de boa qualidade para histórias de usuário que estabelece que elas devem ser pequenas o suficiente para serem concluídas em uma única iteração. Isso significa que elas não devem ser tão grandes ou complexas a ponto de não poder ser concluídas em um único ciclo de desenvolvimento. Histórias muito grandes devem ser divididas para que elas possam ser desenvolvidas e concluídas mais facilmente. Além disso, as histórias de usuário também devem ser pequenas o suficiente para que possam ser priorizadas de forma eficiente.

O gabarito definitivo foi errado, então eu presumo que a banca entenda que não se trata de um limite de tempo de duração específico e, sim, um limite de tempo de duração pequeno (um limite específico poderia ser um limite grande, por exemplo, o que não satisfaz no critério).

**Gabarito:** Errado

**46. (CESPE / BNB – 2022)** Em uma história de usuário, em que se deseja fazer login com a impressão digital do cliente para o seu acesso à sua conta bancária, um exemplo correto de critério de aceitação é: dado que estou realizando login com minha digital, quando eu colocar o dedo cadastrado no leitor, então consigo acessar minha conta.



### Comentários:

O critério de aceitação de uma história de usuário é um conjunto de condições que devem ser atendidas para que uma história de usuário seja considerada concluída. Estas condições podem incluir testes de aceitação, listas de verificação, requisitos de qualidade, documentação e outros critérios estabelecidos pelo time. O critério de aceitação deve ser acordado e documentado pelo time antes do início do desenvolvimento da história.

Dito isso, a questão está perfeita! Esse critério de aceitação define que, se o usuário estiver fazendo login com a impressão digital, então ele deve conseguir acessar sua conta ao colocar o dedo cadastrado no leitor de impressão digital. Isso significa que a função de login com impressão digital está funcionando corretamente e que o usuário pode acessar sua conta mediante o uso desta funcionalidade.

**Gabarito:** Correto

---

**47.(CESPE / BNB – 2022)** No gerenciamento de requisitos, uma adequada configuração, em particular, de uma especificação tem a propriedade de ser imutável.

### Comentários:

Questão polêmica! O gabarito definitivo foi correto, mas eu acho quase impossível existir uma documentação imutável mesmo com a mais adequada configuração. Para mim, a questão caberia recurso.

**Gabarito:** Correto

---

**48.(CESPE / BANRISUL – 2022)** O levantamento de requisitos com casos de uso é muito eficaz para a elicitação de requisitos não funcionais.

### Comentários:

Não, o levantamento de requisitos com casos de uso é muito eficaz para a elicitação de requisitos não funcionais.

**Gabarito:** Errado

---

**49.(CESPE / BANRISUL – 2022)** Em situações em que alguma das partes interessadas não consiga expressar de forma oral as suas necessidades com clareza, recomenda-se o emprego da técnica da etnografia para o levantamento de requisitos.

### Comentários:



Perfeito! A etnografia é uma técnica de observação que pode ser utilizada para compreender os requisitos sociais e organizacionais, ou seja, entender a política organizacional bem como a cultura de trabalho com objetivo de familiarizar-se com o sistema e sua história.

**Gabarito:** Correto

---

**50. (CESPE / BANRISUL – 2022)** A acessibilidade está relacionada à facilidade com que determinada informação é assimilada por pessoas com alguma deficiência.

**Comentários:**

Perfeito! A acessibilidade está completamente relacionada à facilidade de utilização para pessoas com necessidades especiais.

**Gabarito:** Correto

---

**51. (CESPE / BANRISUL – 2022)** A usabilidade é um atributo de qualidade de um projeto que avalia se ele fornece os recursos que os usuários precisam.

**Comentários:**

Não, a usabilidade é um atributo de qualidade que avalia quão fácil uma interface é de usar.

**Gabarito:** Errado

---

**52. (CESPE / FUNPRES-P-EXE - 2022)** As verificações de validade, consistência e completeza são técnicas fundamentais do processo de validação de requisitos.

**Comentários:**

Perfeito! A equipe de revisão deve verificar cada requisito em termos de consistência (para evitar ambiguidade), bem como verificar os requisitos como um todo em termos de completeza.

**Gabarito:** Correto

---

**53. (CESPE / FUNPRES-P-EXE - 2022)** Dentre as técnicas existentes de elicitação de requisitos baseadas em cenários, os casos de uso são modelos que ajudam a identificar agentes e interações do sistema.

**Comentários:**



Casos de Uso são uma técnica para captar os requisitos funcionais de um sistema. Eles servem para descrever as interações de usuários com o sistema, fornecendo uma narrativa sobre como o sistema é utilizado.

**Gabarito:** Correto

---

**54. (CESPE / FUNPRES-EXE - 2022)** A técnica Quality Function Deployment tem como objetivo traduzir os requisitos técnicos em requisitos do cliente.

**Comentários:**

Na verdade, Quality Function Deployment (QFD) é uma técnica da gestão de qualidade que traduz as necessidades do cliente para requisitos de software. É o contrário do que diz a questão.

**Gabarito:** Errado

---

**55. (CESPE / FUNPRES-EXE - 2022)** O protótipo de interface do usuário é o produto final da técnica de prototipação da engenharia de requisitos.

**Comentários:**

O produto final da engenharia de requisitos é a documento de requisitos.

**Gabarito:** Errado

---

**56. (CESPE / FUNPRES-EXE - 2022)** Brainstorming é uma técnica utilizada para o levantamento de requisitos; para facilitar o registro, essa técnica deve ser feita por meio de questionário.

**Comentários:**

Brainstorming é uma técnica que utiliza uma abordagem informal, logo ela não pode ser feita por meio de questionários. Lembrando: Brainstorming é uma abordagem de elicitação ocorrida em grupo em ambientes informais durante cerca de 15 minutos em que toda a ideia deve ser levada em consideração.

**Gabarito:** Errado

---

**57. (CESPE / Petrobrás - 2022)** Ferramentas automatizadas para armazenamento de requisitos, gerenciamento de mudanças e gerenciamento de rastreabilidade são indicadas para apoio ao processo de gerenciamento de requisitos.

**Comentários:**



De acordo com Sommerville, o gerenciamento de requisitos precisa de apoio automatizado, sendo que esse apoio é provido por meio de ferramentas automatizadas que são escolhidas durante a fase de planejamento. Desse modo, as ferramentas de apoio são úteis para o armazenamento de requisitos, o gerenciamento de mudanças e o gerenciamento de rastreabilidade.

**Gabarito:** Correto

**58. (CESPE / Petrobrás - 2022)** Histórias de usuário são ferramentas para a definição de escopo de produtos de software voltadas a fornecer uma análise detalhada sobre a atividade do usuário e a viabilizar a retenção de conhecimento em longo prazo.

#### **Comentários:**

As Histórias de Usuário (User Stories) são artefatos de desenvolvimento utilizados em sistemas geridos segundo metodologias ágeis. Elas são uma descrição resumida de alguma funcionalidade fornecida pelo sistema do ponto de vista de um usuário desse sistema. Importante ressaltar que se trata de uma descrição de uma pequena parte de uma funcionalidade, e não de uma funcionalidade completa. Vejamos, portanto, que a questão não se trata da definição das histórias de usuário.

**Gabarito:** Errado

**59. (CESPE / Petrobrás - 2022)** Os critérios de aceitação descrevem um conjunto mínimo de requisitos que precisam ser atendidos para que valha a pena implementar uma solução específica.

#### **Comentários:**

Exato! Os critérios de aceitação estão relacionados aos requisitos que devem ser atendidos para que seja possível executar um projeto. Esses critérios são as condições, normalmente dadas pelo cliente, que trazem os requisitos mínimos que devem ser alcançados.

**Gabarito:** Correto

**60. (CESPE / Petrobrás - 2022)** Entrevistas e questionários são técnicas comumente usadas para obter informações relacionadas às necessidades de grupos de usuários representados por personas, que exemplificam como um usuário típico interage com um produto.

#### **Comentários:**

Uma persona é uma pessoa fictícia que está interessada no sistema. Dessa forma, é comum utilizar-se entrevistas e questionários para se extrair de tais pessoas suas necessidades relacionadas a um produto, pois isso irá exemplificar como será a interação entre eles.





**Gabarito:** Correto

**61. (CESPE / Petrobrás - 2022)** No contexto de storytelling, é fundamental mitigar as possibilidades de navegação por meio das interfaces e impor à experiência do usuário o sequenciamento estrito das atividades que constituem a sua história.

**Comentários:**

Storytelling trata-se do ato de contar histórias. Dessa forma, pode-se transmitir conhecimentos e compartilhar experiências. Dessa forma, Storytelling nada tem a ver com mitigar possibilidades ou impor experiências aos usuários.

**Gabarito:** Errado

**62. (CESPE / TJ-RJ - 2021)** Na engenharia de requisitos, por estar mais aderente às características dessa técnica, a etnografia é recomendada:

- a) na elicitação da forma como o fluxo dos processos deveria ser feito.
- b) na descoberta dos requisitos organizacionais.
- c) quando se deseja obter uma visão do funcionamento do sistema na forma prevista, independentemente das interferências de seu contexto.
- d) na descoberta de requisitos derivados do conhecimento das atividades de outras pessoas que realizam trabalhos adjacentes ao analisado.
- e) como uma alternativa aos casos de uso para a descoberta dos requisitos explícitos.

**Comentários:**

O principal objetivo da etnografia é que ela ajuda a descobrir requisitos de sistema implícitos, que refletem os processos reais, em vez de os processos formais, onde as pessoas estão envolvidas. São os requisitos derivados da maneira como as pessoas realmente trabalham, em vez da maneira pelas quais as definições de processo dizem como elas deveriam trabalhar. Além disso, essa abordagem é recomendada para levantar requisitos derivadas da cooperação e interação entre pessoas em seu ambiente natural. A redação é meio estranha, mas foi isso que o item (d) quis dizer com descoberta de requisitos derivados do conhecimento das atividades de outras pessoas que realizam trabalhos adjacentes ao analisado.

**Gabarito:** Letra D

**63. (CESPE / TJ-RJ - 2021)** Para os propósitos da modelagem dos requisitos com base em cenários, um suporte apropriado é o uso de

- a) diagrama de casos de uso e histórias de usuários.
- b) diagrama de sequência e diagrama de atividades.



- c) diagramas que representem eventos ou estados.
- d) diagrama de classes e histórias de usuário.
- e) modelagem com cartões CRC e casos de uso.

#### Comentários:

Os diagramas de casos de uso e as histórias de usuários são técnicas baseadas em cenários para a obtenção de requisitos, que identificam os atores envolvidos em uma interação e que descrevem a interação em si.

**Gabarito:** Letra A

**64.(CESPE / TELEBRÁS - 2021)** No âmbito da engenharia de software, o principal produto da engenharia de requisitos é o documento de especificação de requisitos de software.

#### Comentários:

Perfeito! De acordo com Sommerville, o objetivo da engenharia de requisitos é criar e manter um documento de requisitos de sistema.

**Gabarito:** Correto

**65.(CESPE / TCE-RJ – 2021)** O gerenciamento de requisitos trata do desenvolvimento de software por meio da metodologia ágil; isso permite o isolamento entre o desenvolvedor e o usuário, já que é comum ocorrer problema de mudanças de requisitos ao longo do curso do projeto devido ao interfaceamento do usuário com o desenvolvedor.

#### Comentários:

O gerenciamento de requisitos é o processo responsável por compreender, acompanhar e controlar as mudanças nos requisitos de um sistema, identificando as inconsistências – não existe essa relação direta com metodologia ágil. Além disso, não é recomendável haver um isolamento entre o desenvolvedor e o usuário – e os requisitos mudarem não é necessariamente um problema.

**Gabarito:** Errado

**66. (CESPE / TCE-RJ – 2021)** Em um processo de desenvolvimento de software, a elicitacão de requisitos serve para identificar os fatos que compõem os requisitos do sistema.

#### Comentários:

Elicitar significa descobrir, identificar, extrair. A fase Elicitacão, portanto, trata do processo de levantamento e derivação de requisitos de sistema por meio de diversas técnicas.



**Gabarito:** Correto

---

**67. (CESPE / Ministério da Economia – 2020)** Um dos princípios em que se baseia a técnica de dinâmica de grupo conhecida como brainstorm é o atraso de julgamento, que possibilita a geração de muitas ideias antes de se decidir por uma.

**Comentários:**

O atraso de julgamento está relacionado à criação de ideias sem que se importe se tal ideia é boa ou ruim, ou seja, sem fazer um julgamento da ideia. É o que, de fato, se busca em uma tempestade de ideias: gerar um grande número de ideias, sem se preocupar com elas.

**Gabarito:** Correto

---

**68. (CESPE / Ministério da Economia – 2020)** Os requisitos do software mudam com frequência, mas é sempre possível acomodá-los no sistema, pois o software é flexível.

**Comentários:**

A fase da Engenharia de Requisitos é de extrema importância no desenvolvimento de um software. Erros nesse estágio podem ocasionar grandes problemas nas etapas posteriores do projeto e na implementação do sistema. Dessa forma, é um erro afirmar que é sempre possível acomodar novos requisitos.

**Gabarito:** Errado

---

**69. (CESPE / Ministério da Economia – 2020)** Requisitos funcionais envolvem as características de confiabilidade e de desempenho de um sistema.

**Comentários:**

Na verdade, requisitos não-funcionais, que incluem características como confiabilidade, segurança, usabilidade, performance, custos, robustez, etc.

**Gabarito:** Errado

---

**70. (CESPE / Ministério da Economia – 2020)** Elicitar requisitos não inclui somente necessidades dos usuários, mas também extrair informações que surgem de padrões organizacionais, governamentais e industriais em geral, para atender necessidades.

**Comentários:**



Perfeito! Por exemplo: uma técnica usada na Elicitação de Requisitos é a de Etnografia, que é utilizada para compreender os requisitos organizacionais e sociais. Além disso, a Leitura de Documentos é outro exemplo – ela coleta informações como: histórico da organização, cultura e hábitos internos, relacionamentos setoriais, informações financeiras e direcionamentos futuros.

**Gabarito:** Correto

---

**71. (CESPE / TJ-AM – 2019)** A validação dos requisitos exclui diversas considerações, entre elas, a que verifica o impacto da implementação dos requisitos identificados sobre o orçamento do sistema.

#### Comentários:

Nós vimos que durante o processo de validação de requisitos, devem ser realizadas verificações nos requisitos do documento de requisitos, que incluem as verificações de realismo:

**Verificações de realismo:** usando o conhecimento da tecnologia existente, os requisitos devem ser verificados quanto a se realmente podem ser implementados. Essas verificações também devem levar em consideração o **orçamento** e o prazo para o desenvolvimento do sistema – caiu questão de prova recente sobre isso! Logo, o impacto orçamentário é considerado.

**Gabarito:** Errado

---

**72. (CESPE / TJ-AM – 2019)** Na gerência de requisitos, as mudanças no documento de requisitos devem aumentar as referências a outros documentos e aprimorar a interdependência entre suas próprias seções.

#### Comentários:

Na gerência de requisitos, os requisitos devem ser rastreáveis. Não há nenhuma obrigação de se aumentar as referências a outros documentos.

**Gabarito:** Errado

---

**73. (CESPE / TJ-AM – 2019)** Uma especificação de requisitos é inconsistente quando, por exemplo, em um de seus subconjuntos conste que o pagamento será feito antes do fechamento da compra e, em outro subconjunto, conste que o pagamento será feito depois do fechamento da compra.

#### Comentários:



Perfeito! Isso atenta contra a consistência dos requisitos. Lembrem-se de que requisitos de devem ser claros, não-ambíguos, fáceis de entender, completos e **consistentes** – logo, não podem ser ambíguos ou contraditórios.

**Gabarito:** Correto

---

**74. (CESPE / TJ-AM – 2019)** Em um protótipo para validar os requisitos de um software, é admissível deixar de fora os requisitos não funcionais ou reduzir seus padrões.

**Comentários:**

Perfeito! Um protótipo é um modelo ou esboço de funcionalidades, logo é razoável admitir que requisitos não funcionais estejam fora de seu escopo assim como uma redução em seus padrões de qualidade.

**Gabarito:** Correto

---

**75. (CESPE / SLU-DF – 2019)** A interoperabilidade entre um software que esteja em desenvolvimento e outros sistemas existentes é considerada um requisito funcional.

**Comentários:**

A interoperabilidade é evidentemente um requisito não-funcional, uma vez que trata de uma restrição de uma funcionalidade.

**Gabarito:** Errado

---

**76. (CESPE / STM – 2018)** Requisitos de domínio são relativos ao que o sistema deve fornecer, como ele deve reagir a entradas específicas e se comportar em determinadas situações, enquanto os requisitos funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema.

**Comentários:**

Requisitos de Domínio podem ser funcionais ou não-funcionais. Restrições nos serviços ou funções oferecidas pelo sistema são Requisitos Não-Funcionais.

**Gabarito:** Errado

---

**77. (CESPE / STM – 2018)** O processo de verificação visa assegurar que o sistema atende as expectativas e necessidades do cliente por meio da utilização de técnicas de entrevista como brainstorming, grupos focais ou Delft, a partir das quais são extraídos os requisitos não funcionais.



### Comentários:

Quem visa assegurar que o sistema atenda as expectativas e necessidades do cliente é o processo de validação e, não, verificação. Além disso, essas são técnicas de levantamento de requisitos (funcionais).

**Gabarito:** Errado

---

**78.(CESPE / CGM-PB – 2018)** A atividade de gerência de requisitos é a responsável por garantir que mudanças nos requisitos sejam feitas de maneira controlada e documentada, administrando os relacionamentos entre os requisitos e as dependências entre o documento de requisitos e os demais artefatos produzidos no processo de software.

### Comentários:

Galera, que item maravilhoso! É a definição perfeita e impecável da Gerência/Gestão de Requisitos. Ele realmente é o responsável por garantir que mudanças nos requisitos sejam feitas de maneira controlada e documentada, administrando os relacionamentos entre os requisitos e as dependências entre o documento de requisitos e os demais artefatos produzidos no processo de software.

**Gabarito:** Correto

---

**79.(CESPE / ABIN – 2018)** Ao se aplicar a rastreabilidade bidirecional, é possível determinar se todos os requisitos-fonte foram completamente tratados e se todos os requisitos do produto atendem aos requisitos do cliente.

### Comentários:

Esse é um conceito trazido pelo livro do Kechi Hirama, que afirma que a rastreabilidade bidirecional auxilia a determinar se todos os requisitos fonte foram completamente tratados e se todos os requisitos de mais baixo nível podem ser rastreados para uma fonte válida. Eu sei, é realmente complicado ficar estudando bibliografias nada consagradas :(

**Gabarito:** Errado

---

**80.(CESPE / ABIN – 2018)** Definir e manter matriz de rastreabilidade dos requisitos permite controlar e tratar as mudanças em requisitos durante o processo de elicitação e especificação do produto.

### Comentários:





Definir e manter matriz de rastreabilidade dos requisitos realmente permite controlar e tratar as mudanças em requisitos, no entanto ela ocorre durante a atividade de gerenciamento de requisitos e, não, elicitação e especificação de requisitos.

**Gabarito:** Errado

---

**81.(CESPE / ABIN – 2018)** Para que os requisitos sejam refinados e sejam gerados modelos de análise e projeto para codificação, apenas a avaliação e a aprovação por parte do cliente — mesmo após o entendimento dos requisitos — não são suficientes.

#### **Comentários:**

Cuidado com esse tipo de questão! Ela parece ser referente ao Pressman ou Sommerville, mas trata do MPS.BR. O Processo de Gerência de Requisitos (GRE) afirma literalmente que a avaliação e aprovação por parte do cliente após o entendimento dos requisitos por si só não é suficiente para que os requisitos sejam refinados e refletidos em modelos de análise e projeto para a codificação

**Gabarito:** Correto

---

**82.(CESPE / ABIN – 2018)** De acordo com as técnicas facilitadoras de especificação de aplicação, recomenda-se que a descrição de requisitos e regras seja feita diretamente pela equipe técnica, sem a participação do cliente.

#### **Comentários:**

Na verdade, é obrigatória e fundamental a participação do cliente na especificação de requisitos.

**Gabarito:** Errado

---

**83.(CESPE / ABIN – 2018)** No processo de elicitação de requisitos, há atividades relacionadas a identificação, rastreabilidade e mudanças em requisitos.

#### **Comentários:**

Na verdade, rastreabilidade e mudança de requisitos são atividades relacionadas ao Gerenciamento de Requisitos e, não, Elicitação de Requisitos.

**Gabarito:** Errado

---

**84.(CESPE / EBSEH – 2018)** Na especificação de requisitos, são estabelecidos uma escala de medição e os valores aceitáveis para cada requisito de usuário, tornando-o mensurável, ou seja, adicionando a ele um critério de aceitação.



### Comentários:

No livro *Mastering the Requirements Process* (Robertson; Robertson, 2006), temos que: *“Assim, na descrição de requisitos de usuário pode ser suficiente capturar a intenção e depois, na especificação de requisitos de sistema, transformar essa intenção em um requisito mensurável, adicionando a ele um critério de ajuste. É muito comum que, neste processo, um requisito não funcional de usuário dê origem a vários requisitos não funcionais de sistema”*.

Esse critério de ajuste seria o que a questão chama da escala de medição. No entanto, essa não é uma bibliografia consagrada – eu não gosto desse tipo de questão!

**Gabarito:** Correto

---

**85.(CESPE / EBSERH – 2018)** Requisitos externos são derivados de metas, políticas e procedimentos das organizações, do cliente e do desenvolvedor e incluem requisitos de processo, requisitos de implementação, restrições de entrega e restrições orçamentárias.

### Comentários:

Metas, políticas e procedimentos são requisitos organizacionais e, não, externos. Ademais, requisitos externos não incluem requisitos de processo, de implementação, de entrega e restrições orçamentárias – os requisitos externos são regulatórios, éticos ou legais.

**Gabarito:** Errado

---

**86. (CESPE / IPHAN – 2018)** Tanto a etnografia quanto o protótipo podem ser utilizados para validação e elicitação de requisitos, contudo a aplicação de um elimina a possibilidade de uso do outro no mesmo cenário, pois se tratam de técnicas excludentes.

### Comentários:

Elas não são excludentes – pelo contrário, elas são complementares. A etnografia é utilizada para capturar requisitos que o cliente não consegue descrever com clareza; já o protótipo poderia ser utilizado para validar os requisitos descobertos na etnografia, por exemplo.

**Gabarito:** Errado

---

**87.(CESPE / IPHAN – 2018)** A validação de requisitos se sobrepõe à análise de requisitos, pois tem a finalidade de encontrar eventuais problemas nos requisitos e validá-los conforme as necessidades dos usuários do sistema.

### Comentários:



Perfeito! Ela se sobrepõe no sentido de que ela é capaz de descobrir problemas com requisitos quanto à ambiguidade, conflitos, contradições, entre outros – mostrando que os requisitos obtidos definem realmente o sistema que o usuário necessita. Esta fase sobrepõe à análise, pois compreende a descoberta de problemas com os requisitos. A atividade de validação de requisitos é fundamental para o processo de engenharia de requisitos porque ela procura diminuir os custos com a identificação de erros no documento de requisitos, quando eles são encontrados apenas na fase de desenvolvimento do sistema ou quando o sistema já está em operação.

**Gabarito:** Correto

**88. (CESPE / IPHAN – 2018)** Situação hipotética: Como forma de obter os requisitos de apoio para desenvolver um sistema a ser implementado em determinado setor de uma organização, um analista propôs que se observasse o trabalho do dia a dia, anotando-se as tarefas realizadas no referido setor. Assertiva: Para o cenário proposto, é ideal a utilização da técnica de caso de uso alinhada à entrevista.

#### Comentários:

*Observasse o trabalho do dia a dia? Anotando as tarefas realizadas no setor? Só pode estar se referindo à técnica de etnografia e, não, entrevista.*

**Gabarito:** Errado

**89. (CESPE / IPHAN – 2018)** Situação hipotética: Na metodologia de desenvolvimento de software customizada para uma organização, o analista propôs o uso da prototipação na fase de engenharia de requisitos, contudo julgou inviável a utilização da prototipação na fase de projeto de sistemas. Assertiva: Nessa situação, a proposta do analista está incorreta, pois a prototipação tanto pode ser utilizada no processo de engenharia de requisitos, para ajudar na elicitação de requisitos, quanto no projeto de sistema, para apoiar o projeto de interface de usuário.

#### Comentários:

Vamos lá! O examinador foi bastante descuidado. É possível entender que ele quis dizer que realmente é possível utilizar a prototipação tanto na fase de requisitos quanto de projeto. No entanto, ele afirma que a proposta do analista está incorreta porque a prototipação tanto pode ser utilizada no processo de engenharia de requisitos, para ajudar na elicitação de requisitos, quanto no projeto de sistema, para apoiar o projeto de interface de usuário. Ora, não há nenhuma relação direta de causalidade. O fato de o analista julgar **inviável** a utilização da prototipação na fase de projeto de sistemas em nenhum momento implica que ele tenha feito essa avaliação por achar que não é possível aplicá-la – pode ser por achar que o custo seja alto, seja muito trabalho, entre outras possibilidades. Dessa forma, discordo veementemente do gabarito definitivo!



**Gabarito:** Correto

---

**90. (CESPE / PF – 2018)** No desenvolvimento de um sistema de informação, a fase de levantamento de requisitos consiste em compreender o problema, dando aos desenvolvedores e usuários a mesma visão do que deve ser construído para resolvê-lo, e a fase de projeto consiste na realização da descrição computacional, incluindo a arquitetura do sistema, a linguagem de programação utilizada e o sistema gerenciador de banco de dados (SGBD) utilizado.

**Comentários:**

De fato, a fase de levantamento de requisitos consiste em compreender o problema dando aos desenvolvedores e usuários a mesma visão do que deve ser construído. *Mesma visão, Diego? Sim,* aqui visão está sendo utilizado em um sentido amplo como o de um Documento de Visão. Por fim, lembrem-se que – no projeto – nós desenhamos a solução para o problema, já incluindo a arquitetura do sistema, linguagem de programação, sistema gerenciador de banco de dados, etc.

**Gabarito:** Correto

---

**91. (CESPE / MPE-PI – 2018)** A análise de requisitos consiste na área responsável pela identificação das reais necessidades dos clientes de TI. Por meio da análise de requisitos, em conjunto com o cliente, é possível construir uma solução que atenda essas necessidades e desenvolver os requisitos funcionais elencados.

**Comentários:**

Questão simples! A análise de requisitos realmente permite identificar as necessidades dos clientes e construir uma solução que atenda essas necessidades.

**Gabarito:** Correto

---

**92. (CESPE / MPE-PI – 2018)** Situação hipotética: Ao se iniciar a especificação de requisitos de um software para controlar o gasto de folhas impressas de um setor, o analista de requisitos, juntamente com o gestor, definiu um cenário de teste em que, ao se comandar a impressão, a chave do usuário autenticado no sistema que comandar uma impressão acionará o contador de impressões do setor de locação desse usuário. Assertiva: Nessa situação, o teste validará o cenário do requisito definido junto com o gestor.

**Comentários:**

Perfeito! Lembrem-se de que uma das técnicas de validação de requisitos é a geração de casos de teste. Logo, nesse caso, o teste validará o cenário do requisito especificado inicialmente junto ao gestor.



**Gabarito:** Correto

**93.(CESPE / SE-DF – 2017)** Para auxiliar na gerência de riscos e prevenir insatisfações das partes interessadas, deve-se dificultar as modificações na especificação dos requisitos.

**Comentários:**

Hahaha... essa questão é engraçada! *Como assim, cara? Você vai dificultar as modificações na específica dos requisitos do produto de um cliente?* Imaginem vocês chegarem em um pedreiro e falarem: "Amigão, eu tinha falado que queria essa lâmpada aqui, mas eu mudei de ideia e agora eu quero ela ali". E o pedreiro dificultar a modificação que você quer fazer na sua própria casa. Isso não faz sentido – você é o cliente! *E como isso previne insatisfação das partes interessadas?* Com certeza, você vai ficar irritadíssimo!

**Gabarito:** Letra E

**94.(CESPE / SE-DF – 2017)** Um dos objetivos da engenharia de requisitos é integrar tarefas, técnicas, orientações, responsabilidades e papéis em fluxos de trabalho.

**Comentários:**

Isso foi retirado do livro Engenharia de Requisitos: Software Orientado ao Negócio, de Carlos Eduardo Vazquez e Guilherme Siqueira. Segue o trecho: "A Engenharia de Requisitos facilita a interação com o cliente em termos de identificar e entender suas necessidades e na obtenção de um acordo da solução que será entregue. Ela descreve e integra tarefas, técnicas, orientações, papéis e responsabilidade em fluxos de trabalho que: tem início com o entendimento da necessidade do cliente; e passam pelo acordo sobre a solução que será construída".

Galera, vou ser sincero com vocês! Eu errei esse item – achei essa descrição absurdamente abstrata e genérica. No entanto, lendo no contexto do livro, faz todo sentido mesmo. Não se martirizem, caso tenham errado essa questão :P

**Gabarito:** Correto

**95.(CESPE / SE-DF – 2017)** É comum que uma especificação de requisitos inclua as interfaces externas do software.

**Comentários:**

Isso foi retirado do livro Engenharia de Requisitos: Software Orientado ao Negócio, de Carlos Eduardo Vazquez e Guilherme Siqueira. Segue o trecho: "Lista de Requisitos Funcionais: descreve tarefas e serviços que serão fornecidos pelo sistema aos seus usuários (Exemplo: lista de casos de uso, histórias do usuário). Incluir também as interfaces externas do software".



E isso realmente faz todo sentido. A especificação de requisitos deve contemplar as interfaces externas do software.

**Gabarito:** Correto

96. (CESPE / TRE-PE – 2017) No contexto da análise de requisitos, confiabilidade e usabilidade são atributos de qualidade classificados como:

- a) requisitos funcionais.
- b) requisitos de domínio.
- c) requisitos não funcionais.
- d) dependências.
- e) regras de negócio.

**Comentários:**



A questão trata dos requisitos não-funcionais.

**Gabarito:** Letra C

97. (CESPE / TCE-PR – 2016) Com relação aos requisitos de software, assinale a opção correta.

- a) O documento de especificação de requisitos é um documento restrito à equipe de desenvolvimento de software.
- b) As necessidades do usuário são informações que substituem os requisitos do software.
- c) Os requisitos de produto e os requisitos organizacionais são tipos de requisitos funcionais.
- d) Os requisitos funcionais descrevem as funcionalidades, os recursos e as características do software.





e) Os requisitos não funcionais referem-se diretamente às características do software.

#### Comentários:

(a) Errado, é um documento que abrange qualquer parte interessada (inclusive, serve como contrato); (b) Errado, elas não substituem, na verdade elas substanciam os requisitos do software; (c) Errado, na verdade, são requisitos não-funcionais; (d) Correto, eles descrevem funcionalidades, recursos e características do software; (e) Errado, esses são os requisitos funcionais.

**Gabarito:** Letra D

---

**98. (CESPE / TRT-PR – 2016 – Letra D)** Durante a fase de levantamento de requisitos para a construção de um software, compete aos desenvolvedores organizar as necessidades em ordem de prioridade.

#### Comentários:

Na verdade, não compete aos desenvolvedores – compete aos usuários. Eles devem organizar a prioridade das necessidades.

**Gabarito:** Errado

---

**99. (CESPE / TRT-PR – 2016 – Letra E)** O QFD (quality function deployment) identifica como requisitos fascinantes os recursos que extrapolam as expectativas dos clientes.

#### Comentários:

Requisitos Fascinantes são recursos que vão além da expectativa dos clientes e demonstram ser muito satisfatórios quando presentes. Por exemplo, o software para um novo celular vem com recursos-padrão, mas junto vem um conjunto de capacidades não esperadas. Exemplos de Requisitos Fascinantes: tecla multitoque e correio de voz visual.

**Gabarito:** Correto

---

**100. CESPE / MPOG-ATI – 2015)** Tão logo exista uma versão do documento de requisitos, o processo de gerenciamento de requisitos deverá ser iniciado.

#### Comentários:

O processo de gerenciamento de requisitos deve se iniciar assim que uma versão inicial do documento de requisitos esteja disponível, mas o planejamento das mudanças de requisitos deve ser iniciado durante o processo de elicitação de requisitos. A evolução de requisitos, durante o



processo de engenharia de requisitos e após a entrada de um sistema em operação, é inevitável. Logo, a questão está corretíssima.

**Gabarito:** Correto

**101. (CESPE / MPOG-ATI – 2015)** As informações de rastreabilidade de requisitos possibilitam a realização de estimativa do custo de mudanças em requisitos.

#### Comentários:

A rastreabilidade permite avaliar impacto e possibilita estimar o custo de mudanças em requisitos. Galera, observem que a questão utiliza '*possibilita*'. Sempre que a banca faz isso, ela está te ajudando, porque para essa questão estar errada, não deveria haver absolutamente nenhuma possibilidade de estimar custos de mudanças em requisitos por meio de informações de rastreabilidade.

**Gabarito:** Correto

**102. (CESPE / MPOG-ATI – 2015)** As mudanças de requisitos em processos ágeis de desenvolvimento não seguem um processo formal de gerenciamento de requisitos.

#### Comentários:

De fato, as mudanças de requisitos em processos ágeis de desenvolvimento seguem um processo mais informal de gerenciamento. Não há, por exemplo, documentação extensa ou matrizes de rastreabilidade! Em geral, metodologias ágeis não recomendam a utilização de processos muito formais.

**Gabarito:** Correto

**103. (CESPE / MPOG-ATI – 2015)** Para a elicitação dos requisitos, é indicada à empresa a realização de um workshop de requisitos, em que seja determinado um facilitador, mesmo que sem grande experiência com os processos de gerenciamento de requisitos.

#### Comentários:

Essa questão foi bastante polêmica! Alguns podem argumentar que não é técnica mais indicada. No entanto, a questão não afirma isso, ela apenas afirma que é indicada – e, de fato, ela é indicada para elicitação de requisitos. Além disso, conforme eu disse em aula, o facilitador deve ser neutro e responsável por atividades de logística, organização, etc.

Muitas vezes, ele não precisa ser sequer um cara da área de tecnologia, pode ser um cara da área de gestão de pessoas, por exemplo. Seu papel é facilitar o workshop, mas – similar ao Scrum Master



no contexto de Gestão de Projetos de Desenvolvimento de Software – não precisa ter nenhuma experiência específica no gerenciamento de requisitos.

A questão poderia até ser passível de recurso se falasse 'sem experiência alguma', mas como ela disse apenas 'sem grande experiência', eu a avalio como correta.

**Gabarito:** Correto

**104. (CESPE / MPOG-ATI – 2015)** Os requisitos não funcionais a serem especificados estabelecerão restrições que devem ser seguidas por todo o sistema da referida empresa, podendo até mesmo levar à necessidade de definição de requisitos funcionais.

#### Comentários:

*Os requisitos não funcionais estabelecem restrições? Sim! Eles podem levar à necessidade de definição de requisitos funcionais? Também! O que a questão quer dizer é que, ao levantar e especificar os requisitos funcionais, pode-se acabar percebendo a necessidade de se estabelecer novos requisitos funcionais (ou modificá-los).*

Percebam outro detalhe: a questão afirma "podendo até" – sempre que a questão usar o verbo *poder*, ela já te ajuda! *Por que?* Porque, para a questão estar errada, não poderia haver nenhuma possibilidade de se chegar à requisitos funcionais a partir da especificação dos requisitos não-funcionais. *Entenderam isso? Caiu de novo a mesma coisa!*

**Gabarito:** Correto

**105. (CESPE / MPOG-ATI – 2015)** A definição de um protótipo para a validação dos requisitos pode tornar o processo de requisitos mais barato e mais simplificado, já que ele vai corresponder à real forma de uso do sistema a ser construído.

#### Comentários:

De novo, de novo e de novo! Observem que a questão afirma que "pode tornar o processo mais barato e simplificado". E, sim, ele pode tornar o processo mais barato e simplificado. No entanto, o protótipo não necessariamente vai corresponder à real forma de uso do sistema a ser construído. Na verdade, o protótipo, em geral, é utilizado para validar requisitos de alto nível, logo ele não vai contemplar diversas funcionalidades que estarão no sistema real. Pressman afirma:

*"Yet, prototyping can be problematic for the following reasons: 1. Stakeholders see what appears to be a working version of the software, unaware that the prototype is held together haphazardly, unaware that in the rush to get it working you haven't considered overall software quality or long-term maintainability. When informed that the product must be rebuilt so that high levels of quality can be*



*maintained, stakeholders cry foul and demand that "a few fixes" be applied to make the prototype a working product. Too often, software development management relents".*

**Gabarito:** Errado

---

**106. (CESPE / MPOG-ATI – 2015)** Uma forma de validação dos requisitos é a geração de casos de teste para os requisitos documentados.

**Comentários:**

Existem três técnicas de validação de requisitos: Revisão de Requisitos, Prototipação e Geração de Casos de Teste.

**Gabarito:** Correto

---

**107. (CESPE / MPOG-ATI – 2015)** No ciclo de vida do software, o congelamento dos requisitos do software garante que este, quando em desenvolvimento, atenda à expectativa do usuário, desde que tudo que tenha sido requisitado seja implementado.

**Comentários:**

Requisitos não são estáticos, são dinâmicos e precisam ser refinados constantemente. O processo de definição de requisitos gera um feedback que pode acabar modificando os próprios requisitos. Dessa forma, é evidente que o congelamento de requisitos não garante o atendimento à expectativa do usuário. Em geral, usuários não sabem o que querem; aqueles que sabem, mudam de opinião durante o processo de desenvolvimento de software. Logo, mesmo que tudo que foi requisitado tenha sido implementado, pode não atender às expectativas do usuário, tendo em vista que, logo após o congelamento dos requisitos, o usuário pode muito bem querer modificá-lo.

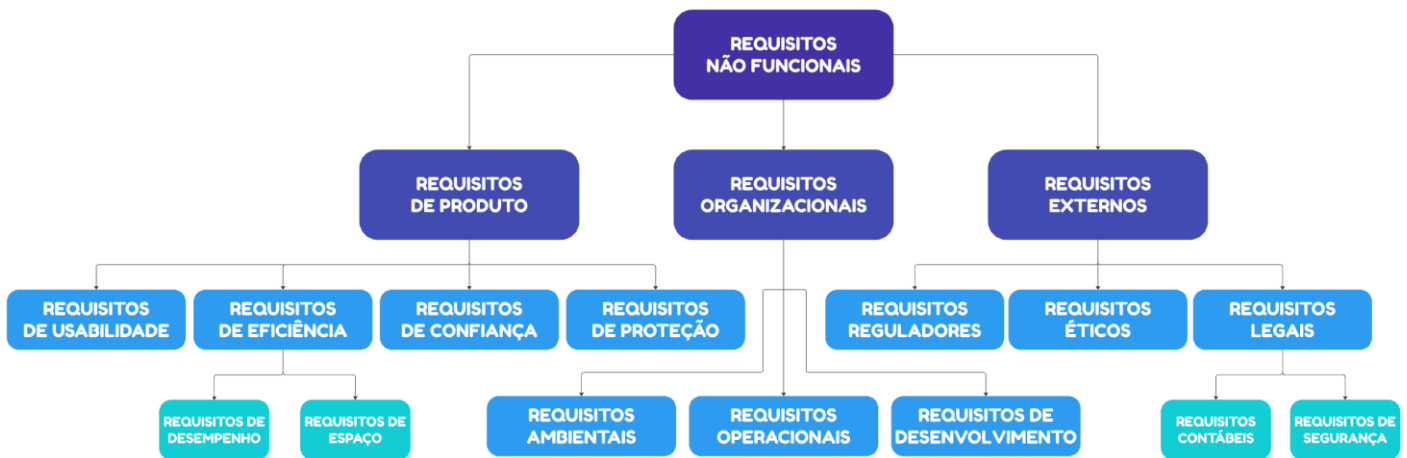
**Gabarito:** Errado

---

**108. (CESPE/ STJ – 2015)** Os requisitos ambientais, operacionais e de desenvolvimento são organizacionais e não funcionais.

**Comentários:**



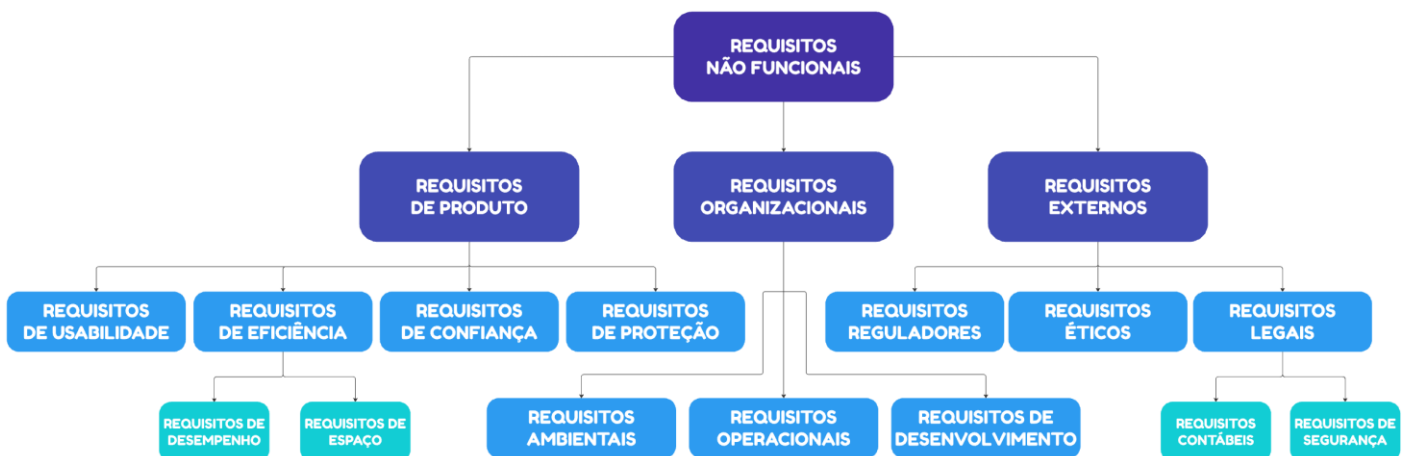


Bastava lembrar da parte superior direita do quadrinho!

**Gabarito:** Correto

109. (CESPE / STJ – 2015) Os requisitos reguladores, legais e éticos são externos e não funcionais.

**Comentários:**



Bastava lembrar da parte inferior do quadrinho! *No entanto, vamos tentar resolver por lógica? Faz sentido pensar em requisitos reguladores, legais e éticos como externos? Sim! E eles são funcionais ou não funcionais? Ora, não-funcionais. Logo, era possível matar a questão!*

**Gabarito:** Correto

110. (CESPE / TJDF – 2015) O uso de protótipo auxilia a descoberta e a validação dos requisitos de software.

**Comentários:**



Perfeito! A prototipação é uma técnica tanto de Elicitação de Requisitos quanto a Validação de Requisitos.

**Gabarito:** Correto

---

**111. (CESPE / TJDFT – 2015)** As técnicas de elicitação e especificação de requisitos incluem a etnografia, a qual é utilizada para compreender os requisitos sociais e organizacionais para determinado projeto.

**Comentários:**

Na verdade, o correto seria: as técnicas de elicitação e ~~especificação~~ análise de requisitos incluem a etnografia, a qual é utilizada para compreender os requisitos sociais e organizacionais para determinado projeto. A banca considerou a questão como correta – não entendo o porquê!

**Gabarito:** Correto

---

**112. (CESPE / TJDFT – 2015)** É caracterizada como requisito funcional a exigência de que, em determinado projeto, o software desenvolvido funcione no sistema operacional Linux, uma vez que essa exigência está diretamente ligada ao software.

**Comentários:**

Trata-se de um requisito não-funcional. *Por que?* Porque funcionar no Sistema Operacional Linux não trata de uma funcionalidade em si, mas uma restrição aos serviços oferecidos.

**Gabarito:** Errado

---

**113. (CESPE / MPU – 2013)** As atividades do gerenciamento de requisitos incluem a análise e a negociação, a qual visa garantir que todos os requisitos do sistema tenham sido declarados de modo não ambíguo, sem inconsistências, omissões e erros.

**Comentários:**

Galera, olhem a bagunça! De acordo com Sommerville, a Análise (junto com a Elicitação) é uma fase da Engenharia de Requisitos. De acordo com Pressman, a Negociação é uma fase da Engenharia de Requisitos. Nenhuma das duas é atividade do Gerenciamento de Requisitos, que também é uma fase da Engenharia de Requisitos (de acordo com Sommerville). Para piorar tudo, quem visa garantir que todos os requisitos tenham sido declarados de modo não ambíguo, sem inconsistências, omissões e erros, é a Validação de Requisitos. Então, está tudo errado!

**Gabarito:** Letra E

---





- 114. (CESPE / MPE-PI – 2012)** Identificada facilidade do cliente em entender uma especificação matemática, é correto utilizar, também, na especificação dos requisitos, notações baseadas em máquinas de estado finito, uma vez que elas podem reduzir a ambiguidade de um documento de requisitos.

**Comentários:**

Bem, se o cliente tem facilidade em entender uma especificação matemática, notações baseadas em máquinas de estado finito serão bastante intuitivas, uma vez que são bastante formais. Para quem não sabe, as máquinas são modelos utilizados para representar diversas situações (Ex: uma lâmpada ligada ou desligada).

**Gabarito:** Correto

---

- 115. (CESPE / EBC – 2011)** No processo de construção e (ou) manutenção de um produto de software, o termo requisito pode ser definido da seguinte forma: uma condição, característica ou capacidade, determinada no universo das necessidades do negócio do usuário, que deve ser atendida por um software na forma de aspectos funcionais e não funcionais.

**Comentários:**

Trata-se de uma boa definição de requisitos. Ele realmente é uma condição, característica ou capacidade, determinada no universo das necessidades do negócio do usuário, que deve ser atendida por um software na forma de aspectos funcionais e não funcionais.

**Gabarito:** Correto

---

- 116. CESPE / EBC – 2011)** O principal artefato elaborado no processo de produção de requisitos do sistema, segundo a ER, é o documento de requisitos. Por sua vez, o documento de requisitos é uma declaração formal dos requisitos para os stakeholders, que podem ser clientes, usuários finais ou a equipe de desenvolvimento do software.

**Comentários:**

Perfeito! Stakeholders são as partes interessadas, dentre elas há os clientes, usuários finais e a própria equipe de desenvolvimento.

**Gabarito:** Correto

---

- 117. (CESPE / BRB – 2011)** O levantamento de requisitos de software privilegia a visão do desenvolvedor em relação aos requisitos de um produto. Já a análise dos requisitos prioriza a visão que o cliente e os usuários têm dos requisitos de um produto.



### Comentários:

Vejam as pegadinhas do CESPE – está invertido! O Levantamento de Requisitos evidentemente privilegia a visão do usuário/cliente; por outro lado, a Análise de Requisitos privilegia a visão do desenvolvedor.

**Gabarito:** Errado

---

**118. (CESPE / FUB – 2011)** A etnografia, uma técnica de levantamento de requisitos, é uma abordagem completa para elicitação, utilizada para compreender os requisitos sociais e organizacionais e que identifica novas características a serem acrescentadas em um sistema.

### Comentários:

Na verdade, ela não apresenta uma abordagem completa! Em geral, essa é uma técnica utilizada em conjunto com outras técnicas. Como ela é uma técnica de observação, isoladamente ela não é muito eficaz na elicitação.

**Gabarito:** Errado

---

**119. (CESPE / EBC – 2011)** Uma das principais técnicas de verificação é a prototipação. Um protótipo é um produto parcialmente desenvolvido, que possibilita aos clientes e desenvolvedores examinarem certos aspectos do sistema proposto e decidir se eles são ou não apropriados ou adequados para o produto acabado.

### Comentários:

A prototipação é uma técnica de **validação de requisitos** e, não, verificação de requisitos.

**Gabarito:** Errado

---

**120. (CESPE / TJ-ES – 2011)** Assim como o software, os requisitos também devem ser avaliados quanto à qualidade. A validação, atividade da engenharia de requisitos, é responsável por garantir que os requisitos tenham sido declarados de forma clara e precisa. Além disso, a validação busca detectar inconsistências, erros e omissões, objetivando alinhar os requisitos às normas estabelecidas para o projeto, produto e processo.

### Comentários:

Os produtos de trabalho criados como consequência da engenharia de requisitos realmente devem ser validados quanto à qualidade durante o passo de validação de requisitos. Esta validação examina a especificação para garantir que todos os requisitos do sistema foram estruturados de maneira não ambígua, que as inconsistências, omissões e erros foram apagados e corrigidos, e que



os produtos de trabalho estão em conformidade com os padrões estabelecidos para o processo, para o projeto e para o produto. Logo, questão perfeita!

**Gabarito:** Correto

---

**121. (CESPE / STM – 2011)** São consideradas técnicas de validação de requisitos: revisões de requisitos, prototipação e geração de casos de teste.

**Comentários:**

De fato, durante a validação de requisitos, uma série de técnicas pode ser utilizada, tais como: Revisão de Requisitos, Prototipação e Geração de Casos de Teste.

**Gabarito:** Correto

---

**122. (CESPE / TJ-ES – 2011)** Verificação e validação são atividades da análise de software, necessárias para se identificar o que o software precisa executar, seguida de uma avaliação do usuário quanto às atividades definidas.

**Comentários:**

Vamos por partes: de fato, são atividades da análise de software (eu disse análise de software e, não, requisitos). Ela faz parte da Engenharia de Requisitos, que busca entender o que o software precisa executar e de uma posterior avaliação do usuário. A redação da questão está bem confusa, pois dá a entender que a verificação identifica o que o software precisa executar.

**Gabarito:** Correto

---

**123. (CESPE / MEC – 2011)** A rastreabilidade de requisitos ocorre apenas na relação entre os requisitos propriamente ditos e os artefatos ou subprodutos de desenvolvimento gerados.

**Comentários:**

A rastreabilidade é a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados. Ela é frequentemente representada por meio de matrizes de rastreabilidade que relacionam os requisitos aos stakeholders, aos outros requisitos, aos módulos de projeto, aos artefatos ou subprodutos, etc. Para responder à questão, bastava lembrar que os requisitos se relacionam também entre si.

**Gabarito:** Errado

---

**124. CESPE / ABIN – 2010)** Requisitos não funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema, e podem ser, também, declarações de serviços que o sistema



deve fornecer, como o sistema deve reagir a entradas específicas e como deve comportar-se em diversas situações.

### Comentários:

A segunda parte da questão está incorreta! Na verdade, declarações de serviços que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve comportar-se em diversas situações, trata-se de requisitos funcionais.

**Gabarito:** Errado

**125. (CESPE / MPU – 2010)** Os requisitos normativos, geralmente oriundos da análise das regras de negócio a que está submetido um sistema, nunca podem ser considerados requisitos funcionais, por estarem fora do sistema, ou seja, do domínio do negócio.

### Comentários:



Bem, a questão disse que nunca pode ser considerado um requisito funcional. De fato, geralmente ele é um requisito não-funcional (como podemos ver na imagem), mas pode haver casos em que esse requisito normativo faça parte do domínio do negócio e deva ser implementado como uma funcionalidade. *Vamos pensar em um exemplo?* Um software de RH de um órgão público deve implementar funcionalidades de acordo com a Lei 8.112/90. Nesse caso, trata-se de um requisito tanto funcional quanto não-funcional (depende do ponto de vista).

**Gabarito:** Errado

**126. (CESPE / ABIN – 2010)** Se os requisitos forem organizados de acordo com os diversos pontos de vista relativos a grupos de usuários do sistema, é possível identificar aqueles comuns a todos ou à maioria dos pontos de vista. Esses requisitos comuns podem estar relacionados a assuntos separados, implementados como extensões da funcionalidade central.

### Comentários:



Essa não precisava nem saber o conteúdo. *Como requisitos comuns podem ser implementados como extensões da funcionalidade central?* Se são requisitos comuns, devem fazer parte da funcionalidade central! Logo, não faz sequer sentido!

**Gabarito:** Errado

---

**127. (CESPE / MPU – 2010)** O levantamento de requisitos é realizado ao final da primeira versão de um protótipo, para se definir, junto aos envolvidos no processo, quais são as premissas básicas para o início do entendimento das funcionalidades desejadas.

**Comentários:**

Essa questão não faz sentido, uma vez que – para se fazer um protótipo – é necessário levantar uma certa quantidade de requisitos! *Como se pode começar a levantar requisitos no final da primeira versão do protótipo?* Também não faz sentido!

**Gabarito:** Errado

---

**128. (CESPE / MPU – 2010)** Embora a criação de uma sequência ilustrada de telas por meio de programas de desenho gráfico seja útil para a identificação de alguns requisitos do software, ela não é considerada uma atividade de prototipação por não envolver o uso de uma linguagem de programação.

**Comentários:**

Na verdade, a identificação de requisitos é independente de linguagem de programação. Logo, essa sequência ilustrada de telas por meio de programas pode – sim – ser considerada uma atividade de prototipação!

**Gabarito:** Errado

---

**129. (CESPE / DETRAN-ES – 2010)** A técnica de brainstorm é adequada para a produção de especificações de requisitos para um sistema de informação em desenvolvimento.

**Comentários:**

A técnica de brainstorm é adequada para a elicitación ou levantamento de requisitos e, não, para a especificação de requisitos.

**Gabarito:** Errado

---



- 130. (CESPE / MPU – 2010)** A verificação de requisitos tem por objetivo analisar se os modelos construídos estão de acordo com os requisitos definidos. Por sua vez, a validação de requisitos visa assegurar que as necessidades do cliente estão sendo atendidas por tais requisitos.

**Comentários:**

A diferença entre Verificação e Validação de Requisitos é, em geral, ignorada em prova – ambos são chamados apenas de Validação de Requisitos. No entanto, via de regra, a verificação de requisitos tem o objetivo de descobrir se os requisitos são claros, precisos, completos e consistentes, e tem por objetivo analisar se os modelos construídos estão de acordo com os requisitos definidos.

Já a Validação de Requisitos se ocupa em mostrar que os requisitos realmente definem o sistema que o cliente deseja, isto é, visa assegurar que as necessidades do cliente estão sendo atendidas por tais requisitos. Logo, questão perfeita!

**Gabarito:** Correto

---

- 131. (CESPE / TCU – 2010)** Por se tratar de função essencial da engenharia de requisitos, a gestão formal de requisitos é indispensável mesmo para projetos de pequeno porte, com apenas duas ou três dezenas de requisitos identificáveis.

**Comentários:**

A gestão formal de requisitos é iniciada somente para grandes projetos. Para pequenos projetos, essa função de engenharia de requisitos é consideravelmente menos formal e, muitas vezes, dispensável.

**Gabarito:** Errado

---

- 132. (CESPE / DETRAN-DF – 2009)** Requisitos funcionais são restrições sobre as funções ou serviços oferecidos pelo sistema. Esses requisitos consideram as declarações de serviços, a forma do sistema reagir e como ele deve se comportar em determinadas situações. Cenários e casos de uso são técnicas eficazes para elicitação de requisitos funcionais segundo pontos de vista de interação.

**Comentários:**

A questão fez uma mistura! Requisitos não-funcionais são restrições sobre as funções ou serviços oferecidos pelo sistema.

**Gabarito:** Errado

---





**133. (CESPE / IPEA – 2009)** Elicitação envolve a identificação sistemática de requisitos nem sempre explicitados pelos clientes. Protótipos, pesquisas estruturadas, testes-beta, análise de casos de negócio, walkthroughs, QFD, grupos de trabalho são exemplos de técnicas utilizadas para eliciar necessidades, expectativas, restrições e interfaces dos stakeholders para todas as fases do ciclo de vida do produto.

#### Comentários:

A definição de elicitação de requisitos está correta – lembrem-se da técnica de etnografia, utilizada para eliciar requisitos implícitos. Um aluno já me perguntou: "*Professor, Testes-beta também?*". De fato, não é nada usual! No entanto, imaginemos a realização de um Teste-beta (aquele que ocorre no ambiente real de utilização do usuário) e descobrimos um problema.

*Esse problema pode se transformar em um novo requisito, concordam? Portanto, eu posso utilizá-lo como uma técnica para descobrir novos requisitos. Porém, é preciso ter aquela experiência de concursado (adquirida quando se faz várias questões e provas) para saber quando esse é o foco da questão e quando não é! Nesse caso, não era o foco da questão.*

**Gabarito:** Correto

**134. (CESPE / TCE-RN – 2009)** A etnografia é uma técnica utilizada para a descoberta de requisitos de sistemas de software na qual, por meio de observações, procura-se compreender os requisitos sociais e organizacionais do ambiente onde o sistema será usado.

#### Comentários:

Essa é realmente a função da técnica de etnografia. Em geral, essa é uma técnica utilizada em conjunto com outras técnicas. Como ela é uma técnica de observação, isoladamente ela não é muito eficaz na elicitação.

**Gabarito:** Correto

**135. (CESPE / IPEA – 2009)** A política organizacional para o planejamento e execução do processo de gerenciamento de requisitos reflete as expectativas organizacionais para processos de gestão de requisitos e para que seja possível identificar inconsistências entre os requisitos e os planos do projeto.

#### Comentários:

A gestão ou gerenciamento de requisitos permite identificar inconsistências entre o projeto inicial e os requisitos. Ele é o processo responsável por compreender, acompanhar e controlar as mudanças dos requisitos de sistema, identificando inconsistências.



**Gabarito:** Correto

---

**136. (CESPE / STJ – 2008)** Os requisitos de um sistema podem ser descrições dos serviços fornecidos ou restrições operacionais. Requisitos podem ainda ser classificados como funcionais, não funcionais, ou de domínio. A engenharia de requisitos visa compreender e definir os requisitos. Um processo de engenharia de requisitos pode envolver o estudo de viabilidade, a análise, a especificação e a validação de requisitos.

**Comentários:**

Vejam que o início é idêntico ao da questão anterior! Sim, algumas bancas copiam outras bancas e algumas bancas copiam a si mesmas. Tudo correto na questão...

**Gabarito:** Correto

---

**137. (CESPE / SERPRO – 2008)** O levantamento de requisitos é importante, porém não é fundamental, pois recomenda-se avançar o quanto antes para as demais atividades que permitam uma visualização do software e reduzam a ansiedade do cliente em ver algo pronto.

**Comentários:**

Galera, podem rir à vontade! Essa questão é hilária... não é fundamental? O Levantamento de Requisitos é importante, fundamental, essencial, básico, determinante, substancial, crucial e indispensável e vários outros sinônimos.

**Gabarito:** Errado

---

**138. CESPE / MPE-RR – 2008)** Os requisitos de um sistema são descrições dos serviços fornecidos pelo sistema e suas restrições operacionais. O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é denominado engenharia de requisitos. Requisitos de um sistema de software podem ser funcionais, não funcionais ou de domínio.

**Comentários:**

Questão muito bem escrita! A definição de requisitos está correta, a definição de engenharia de requisitos está correta e a classificação de requisitos também está correta.

**Gabarito:** Correto

---

**139. (CESPE / MPU – 2007)** A especificação de requisitos permite, em determinado momento, revelar o que o sistema irá realizar no que se refere às funcionalidades, sem definir, nesse momento, como as funcionalidades serão implementadas.



### Comentários:

Ela realmente não revela como as funcionalidades serão implementadas. este tipo de requisito deve simplesmente definir o que o sistema deve fazer e não como ele deve ser implementado, ou seja, os requisitos de sistema devem simplesmente descrever o comportamento externo do sistema e suas restrições operacionais.

**Gabarito:** Correto

---

**140. (CESPE / MPU – 2007)** Na validação de requisitos — parte integrante da especificação desses requisitos —, é correto o uso de diagramas da UML, tais como diagrama de classes, de casos de uso e de interação.

### Comentários:

Galera, questão um pouco polêmica! *Por que?* Porque a Validação de Requisitos não é parte da Especificação de Requisitos – são fases distintas. De todo modo, a questão foi dada como certa! Outra parte que gera alguma dúvida é dizer que é correta a utilização de diagramas da UML! Alguns afirmam que não faz sentido o uso de Diagramas de Classes na validação de requisitos. Galera, de fato, o Diagrama de Casos de Uso é o mais adequado, mas não há nada de incorreto utilizar também o Diagrama de Classes para auxiliar na validação junto ao cliente.

**Gabarito:** Correto

---

**141. (CESPE / SERPRO – 2005)** O gerenciamento de requisitos inclui, entre outras, as seguintes atividades: levantar, analisar, especificar, validar e prototipar requisitos funcionais e não-funcionais.

### Comentários:

A questão trata claramente de engenharia de requisitos e, não, de gerenciamento de requisitos. Eu não sei se não entraram com recurso contra essa questão ou se entraram e o CESPE o ignorou. O fato é que essa questão é absurdamente errada, mas o gabarito foi mantido como correto.

**Gabarito:** Correto

---

**142. (CESPE / SERPRO – 2005)** Uma das principais atividades relacionadas à engenharia de software é o levantamento dos requisitos. Nesse contexto, foi introduzida, na década de 80 do século XX, uma técnica de entrevista conhecida como JAD (Joint Application Development), que consistia em uma rápida entrevista e um processo acelerado de coleta de dados em que todos os principais usuários e o pessoal da análise de sistemas agrupavam-se em uma única e intensiva reunião.



### Comentários:

Galera, eu não sei porque essa questão foi dada como certa! JAD não consiste em uma rápida entrevista, mas em uma intensa reunião. Ademais, não se trata necessariamente de uma única reunião. Acredito que caberia recurso!

**Gabarito:** Correto

---

**143. (CESPE / AGE-ES – 2004)** A engenharia de requisitos fornece mecanismos que permitem entender e analisar a necessidade de o cliente avaliar a exequibilidade, negociar uma solução razoável e especificá-la de maneira não-ambígua, validar a especificação e administrar os requisitos.

### Comentários:

Galera, discordo dessa questão! A redação dela é extremamente confusa – ela diz que o cliente deve avaliar a exequibilidade. Uma redação correta seria: *A engenharia de requisitos fornece mecanismos que permitem entender e analisar a necessidade do cliente, avaliar a exequibilidade, negociar uma solução razoável e especificá-la de maneira não-ambígua, validar a especificação e administrar os requisitos.* Contudo, o CESPE considerou-a como correta.

**Gabarito:** Correto

---

**144. (CESPE / Prefeitura de Boa Vista – 2004)** Requisitos adequadamente definidos constituem base importante sobre a qual um sistema poderá ser bem desenvolvido. No processo de engenharia de requisitos, o estudo de viabilidade utiliza as informações do processo de levantamento de requisitos para gerar um relatório que recomenda se é viável ou não realizar o processo de desenvolvimento do sistema.

### Comentários:

Opa... calma aí! Estudo de Viabilidade vem antes do processo de Levantamento de Requisitos. *Como ele poderia utilizar as informações do processo de levantamento de requisitos?* Impossível!

**Gabarito:** Errado

---

**145. (CESPE / COHAB – 2004)** O QFD (Quality Function Deployment) tem uma abordagem embasada na criação de uma equipe formada por clientes e desenvolvedores, que trabalham juntos para identificar o problema, propor elementos da solução, negociar diferentes abordagens e especificar um conjunto de requisitos da solução.

### Comentários:



Essa questão foi retirada do Pressman:

*“O levantamento de requisitos (também chamado elicitação de requisitos) combina elementos de solução de problemas, elaboração, negociação e especificação. Para estimular uma abordagem colaborativa e orientada a equipes em relação ao levantamento de requisitos, os envolvidos trabalham juntos para identificar o problema, propor elementos de solução, negociar diferentes abordagens e especificar um conjunto preliminar de requisitos de solução”.*

No livro, o autor cita algumas abordagens de levantamento de requisitos, tais como: Coleta Colaborativa de Requisitos e o QFD. A questão trata da primeira abordagem e, não, da segunda – e, por isso, está errada.

**Gabarito:** Errado

**146. (CESPE / COHAB – 2004)** As atividades de análise de requisitos resultam na especificação das características operacionais do software, na indicação da interface do software com outros elementos do sistema e no estabelecimento de restrições que o software deve satisfazer.

#### Comentários:

Essa questão foi retirada do Pressman:

*“A Análise de Requisitos resulta na especificação das características operacionais do software, indica a interface do software com outros elementos do sistema e estabelece restrições a que o software deve atender. Permite ainda que você amplie os requisitos básicos estabelecidos durante as tarefas de concepção, levantamento e negociação, que são parte da engenharia de requisitos”.*

Logo, está perfeita! Atentem-se: a Análise de Requisitos é bastante semelhante à Elaboração, ou seja, expansão, refinamento e detalhamento de requisitos – sob a perspectiva do desenvolvedor e, não, do usuário.

**Gabarito:** Correto

**147. (CESPE / COHAB – 2004)** À medida que os requisitos são elucidados, o analista de software pode criar um conjunto de cenários, ou seja, casos de uso, que identificam uma linha de uso para o sistema a ser construído.

#### Comentários:

Questão linda! Pode-se modelar os requisitos de um sistema por meio da técnica de casos de uso (lembrando que um cenário é um conjunto de casos de uso).

**Gabarito:** Correto







## QUESTÕES COMENTADAS – FCC

1. (FCC / SEFAZ-AP – 2022) Considere as seguintes especificações de requisitos de software:

I. O sistema deve calcular a dívida do contribuinte aplicando a alíquota de 15% quando o lucro ultrapassar o teto de contribuição.

II. O tempo de resposta da consulta à dívida ativa da empresa não deve ultrapassar os 13 ms em situações normais de processamento.

III. O SLA (Acordo de Nível de Serviço) com o contribuinte consulente deve prever jornada de 24 horas/dia × 7 dias por semana.

IV. A tela de consulta à dívida ativa só pode ser acessada mediante login e senha corretos correspondentes àqueles designados ao CNPJ do contribuinte consulente.

Esses requisitos são, correta e respectivamente, dos tipos

- a) funcional, técnico, de usuário e não funcional.
- b) funcional, não funcional, não funcional e funcional.
- c) funcional, técnico, de sistema e não funcional.
- d) não funcional, não funcional, técnico e de sistema.
- e) não funcional, de usuário, técnico e funcional.

### Comentários:

(I) Requisito Funcional, dado que representa uma funcionalidade do sistema; (II) Requisito Não Funcional, dado que se trata de uma restrição a uma funcionalidade do sistema relacionada ao tempo de resposta; (III) Requisito Não Funcional, dado que se trata de uma restrição a uma funcionalidade do sistema relacionada ao acordo de nível de serviço; (IV) Requisito Funcional, dado que se trata de uma funcionalidade do sistema.

**Gabarito:** Letra B

2. (FCC / AL-AP – 2020) Considere a lista abaixo, elaborada durante um levantamento de requisitos na Assembleia Legislativa do Amapá, para um sistema hipotético de avaliações internas:

- 1. Registrar avaliação de colaborador por parlamentar: O sistema deve permitir ao parlamentar, em uma única tela, a avaliação de todos os seus colaboradores.



2. Considerar Aspectos Legais: O sistema deve seguir orientações elencadas na Resolução 099/XXXX do Conselho Legislativo do Estado.
3. Registrar autoavaliação de parlamentar: O sistema deve permitir ao parlamentar sua autoavaliação em relação às disposições legais sob as quais atuou no período.
4. Atentar à Segurança: O sistema deve fornecer mecanismos de segurança e autenticação alinhados com os adotados pelo processo XPTO.
5. Impedir acesso direto ao processo XPTO: O sistema deverá mostrar ao usuário que existem formulários de avaliação a serem respondidos e dará a opção de respondê-los depois.

Adotando RFU para requisitos funcionais e RNF para não-funcionais, a classificação correta e respectiva da lista 1 a 5 acima é:

- a) RFU, RFU, RFU, RNF e RNF.
- b) RFU, RNF, RFU, RNF e RNF.
- c) RFU, RNF, RFU, RNF e RFU.
- d) RNF, RNF, RFU, RNF e RFU.
- e) RNF, RFU, RFU, RNF e RNF.

### Comentários:

(1) Trata-se de um requisito funcional pois descreve o que o sistema deve fazer (RFU); (2) Trata-se de um requisito não-funcional, mais especificamente um Requisito Externo, relacionado a um aspecto legal, que deve ser seguido (RNF); (3) Trata-se de um requisito funcional pois descreve o que o sistema deve fazer (RFU); (4) Trata-se de um requisito não-funcional pois descreve um comportamento do produto. Além disso, também pode ser classificado como um Requisito de Produto (RNF); (5) Trata-se de um requisito funcional pois descreve o que o sistema deve fazer; (RFU).

**Gabarito:** Letra C

**3. (FCC / TRT-19 – 2019)** A Engenharia de Requisitos utiliza algumas técnicas que apoiam as atividades de levantamento de requisitos, sendo a entrevista uma das mais utilizadas. Uma entrevista pode ser estruturada de formas diferentes, como na estrutura em:

- a) diamante, que envolve sessões de workshop com os usuários os quais assumem papéis de documentadores, escrevendo os requisitos em flipcharts.
- b) brainstorming, em que inicia-se com perguntas mais genéricas sobre o sistema e finaliza-se com perguntas mais específicas, sendo geralmente utilizada com usuários que desconhecem o assunto.



- c) funil, na qual procura-se manter o usuário interessado no assunto e para isto utilizam-se perguntas variadas sobre o sistema, sorteadas com um dado.
- d) diamante, na qual os usuários escrevem os requisitos em papel, todos ao mesmo tempo, em uma tempestade de ideias, para estimular requisitos criativos.
- e) pirâmide, em que inicia-se com perguntas mais específicas sobre o sistema e finaliza-se com perguntas mais genéricas, sendo geralmente utilizada com usuários mais relutantes.

### Comentários:

(a) Errado, isso está mais próximo da técnica de Interpretação de Papeis; (b) Errado, essa técnica não é uma estrutura de entrevista e essa descrição trata claramente da estrutura em diamante, porque começa com perguntas mais genéricas e termina com perguntas mais específicas; (c) Errado, essa não é a descrição de uma entrevista do tipo funil, essa é a descrição de uma entrevista não estruturada; (d) Errado, essa não é a descrição de uma entrevista do tipo diamante, essa é uma descrição da técnica de brainstorming; (e) Correto, basta lembrar do formato da pirâmide. Inicia-se com perguntas mais específicas e termina com perguntas mais genéricas.

**Gabarito:** Letra E

4. (FCC / AFAP – 2019) Um Analista de Informática levantou os requisitos para desenvolver um sistema de gestão. Dentre os requisitos levantados,
- I. o sistema deve apresentar a tela de login e senha antes de cada transação e validar o acesso com base nas políticas de segurança organizacional.
  - II. o sistema deve estar disponível para a diretoria em tempo integral, ou seja, 24 x 7.
  - III. o tempo de resposta de uma consulta da alta administração não pode exceder a 5 milissegundos.
  - IV. cada Diretor que usa o sistema deve ser identificado apenas por sua matrícula de cinco dígitos seguidos do código de segurança.
  - V. o sistema deverá gravar um log de autenticação a cada transação completada, contendo a identificação do usuário, data e equipamento utilizado.
  - VI. os backups do sistema deverão ser feitos diariamente a fim de evitar a eventual perda de dados sem capacidade de recuperação.

Contêm um requisito funcional e um requisito não funcional, respectivamente, APENAS os itens



- a) II e I.
- b) V e I.
- c) IV e VI.
- d) II e III.
- e) VI e V.

### Comentários:

Lembremos que os requisitos funcionais especificam o que o software deve fornecer, como deve reagir a entradas específicas e como se comportar em determinadas situações. Já os requisitos não funcionais são restrições aos serviços ou funções oferecidas pelo software (exemplos: desempenho, confiabilidade, segurança, processo de desenvolvimento, etc).

(I) Requisito Funcional, visto que se trata de uma funcionalidade ou serviço oferecido pelo sistema;

(II) Requisito Não-Funcional, visto que se trata de uma restrição de uma funcionalidade do sistema em relação a sua disponibilidade;

(III) Requisito Não-Funcional, visto que se trata de uma restrição de uma funcionalidade do sistema em relação ao tempo de resposta;

(IV) Requisito Funcional, visto que se trata de uma funcionalidade ou serviço oferecido pelo sistema;

(V) Requisito Não-Funcional, visto que se trata de uma restrição de uma funcionalidade do sistema em relação à auditoria e segurança;

(VI) Requisito Não-Funcional, visto que se trata de uma restrição de uma funcionalidade do sistema em relação à integridade e segurança.

**Gabarito:** Letra C

5. (FCC / TRF4 – 2019) Suponha que um Analista de TI, participando da etapa de análise de requisitos de um sistema de emissão de certidão negativa para o TRF4, tenha elencado os requisitos apresentados abaixo:

1. Utilizar interface responsiva para que possa ser executado em dispositivos móveis e na web.
2. Validar o tipo de certidão solicitado.
3. Emitir certidão negativa após verificação de situação do requerente.
4. Solicitar o CPF do requerente.
5. Responder ao clique único do usuário em qualquer botão da interface.
6. Validar o CPF do requerente.
7. Restaurar os dados automaticamente após falhas não programadas.



8. Solicitar o nome do requerente.
9. Oferecer dois tipos de certidão: para fins gerais e para fins eleitorais.
10. Emitir aviso de impossibilidade de emissão da certidão.

Sobre os requisitos, é correto afirmar que:

- a) todos são funcionais.
- b) todos são não funcionais.
- c) 1, 5 e 7 são não funcionais.
- d) apenas 3, 4, 8, 9 e 10 são funcionais.
- e) apenas 2, 6 e 7 são não funcionais.

### Comentários:

(1) RNF; (2) RF; (3) RF; (4) RF; (5) RNF; (6) RF; (7) RNF; (8) RF; (9) RF; (10) RF.

**Gabarito:** Letra C

6. (FCC / SAMASA Campinas – 2019) O diagrama faz referência à QFD –



- a) Quality Function Deployment, uma técnica da gestão de qualidade que traduz as necessidades do cliente para requisitos de software, buscando maximizar a sua satisfação.
- b) Questionário de Funcionalidades para Desenvolvimento, uma técnica para priorização de requisitos que facilita a criação de casos de uso.



- c) Questionnaire For Diagram, uma lista de perguntas que ajudam a entender melhor o problema e permitem que o cliente expresse os requisitos essenciais para a criação de diagramas de caso de uso.
- d) Quality Function Development, uma técnica para priorização de requisitos, especializada para a criação de casos de uso.
- e) Questionário de Funcionalidades para Desenvolvimento, uma técnica da gestão de qualidade que traduz as necessidades do cliente para as funcionalidades a serem incorporadas no software.

### Comentários:

QFD é a sigla para Quality Function Deployment – técnica da gestão de qualidade que traduz as necessidades do cliente para requisitos de *software*, buscando maximizar a sua satisfação.

**Gabarito:** Letra A

7. (FCC / SEFAZ-BA – 2019) Um profissional da área administrativa de certa instituição recebeu um Analista de Sistemas que estava fazendo o levantamento de requisitos para a construção de um novo software. Ao informar ao Analista um requisito não funcional para seu departamento, o profissional corretamente disse que:

- a) a resposta a uma consulta de dados deveria durar no máximo dois segundos para não atrasar seu trabalho.
- b) o sistema deveria permitir a alteração de dados incluídos de forma equivocada.
- c) o acesso ao sistema deveria ser por meio de uma senha composta por letras e números e não apenas por números.
- d) o sistema deveria permitir a exclusão de registros de pessoas que deixaram de ser clientes da instituição.
- e) o sistema, após consultar os dados de um cliente, deveria permitir a impressão dos dados.

### Comentários:

(a) Correto, tempo de resposta é um RNF; (b) Errado, permitir alteração de dados incluídos de forma equivocada é um RF; (c) Errado, o acesso por meio de senha e sua composição é um RF; (d) Errado, permitir exclusão de registros de pessoas que deixaram de ser clientes é um RF; (e) Errado, permitir a impressão de dados é um RF.





**Gabarito:** Letra A

8. (FCC / SEFAZ-BA – 2019) Um Auditor Fiscal da área de Tecnologia da Informação está participando do processo de levantamento de requisitos para o desenvolvimento de um novo software. Os requisitos a seguir foram elencados:
- I. Um usuário deve ser capaz de pesquisar a lista de contribuintes devedores.
  - II. O sistema deve gerar a lista de contribuintes com atendimento agendado naquele dia.
  - III. O sistema deve se adequar às leis que garantem o sigilo das informações.
  - IV. Cada usuário do sistema deverá ser identificado por um número de 8 dígitos.
  - V. O Sistema deve ter suporte para os sistemas operacionais Linux e Windows.
  - VI. A alteração dos dados de um contribuinte só poderá ser concretizada após confirmação.
  - VII. Toda consulta deve retornar os valores solicitados em até 20 segundos.
  - VIII. A gravação dos dados só deverá ser efetuada após o preenchimento de todos os campos de preenchimento obrigatório.
  - IX. Os dados devem ser armazenados em servidores em cluster para garantir a disponibilidade.

São requisitos funcionais os que constam APENAS em:

- a) I, II, III, IV, V, VI e VII.
- b) II, IV, VII e VIII.
- c) I, II, IV, VI e VIII.
- d) IV, VI, VIII e IX.
- e) II, IV, V, VI e VIII.

**Comentários:**

(I) RF; (II) RF; (III) RNF; (IV) RF; (V) RNF; (VI) RF; (VII) RNF; (VIII) RF; (IX) RNF;

**Gabarito:** Letra C

9. (FCC / SEMEF-AM – 2019) Considerando que a Fazenda Municipal emprega o gerenciamento de requisitos, ganha importância o cuidado com os chamados requisitos voláteis, dentre os



quais fazem parte os requisitos que surgem à medida que o cliente vai aprimorando sua compreensão do sistema, denominados requisitos:

- a) mutantes.
- b) de compatibilidade
- c) emergentes
- d) adaptativos
- e) secundários

### Comentários:

A questão trata dos Requisitos Emergentes, que são aqueles que não podem ser completamente definidos quando o sistema é especificado e emergem (olha a dica!) à medida que a compreensão do cliente sobre o sistema se desenvolve.

**Gabarito:** Letra C

**10. (FCC / SEMEF-AM – 2019)** Ao fazer uso da engenharia de requisitos em projetos, deve-se analisar o processo de elicitação e análise de requisitos, o qual pode ser dividido nas seguintes atividades:

- I. Documentação de Requisitos.
- II. Classificação e Organização de Requisitos.
- III. Obtenção de Requisitos.
- IV. Priorização e Negociação de Requisitos.

A ordem sequencial correta para a execução dessas atividades é:

- a) I, III, IV e II.
- b) II, IV, III e I.
- c) III, II, IV e I.
- d) IV, I, II e III.
- e) III, I, II e IV.

### Comentários:

POR SOMMERVILLE	POR PRESSMAN
ESTUDO DE VIABILIDADE	CONCEPÇÃO
ELICITAÇÃO E ANÁLISE DE REQUISITOS	LEVANTAMENTO
OBTENÇÃO DE REQUISITOS	ELABORAÇÃO
CLASSIFICAÇÃO E ORGANIZAÇÃO	
PRIORIZAÇÃO E NEGOCIAÇÃO	NEGOCIAÇÃO
DOCUMENTAÇÃO DE REQUISITOS	





A ordem correta é: (III) Obtenção de Requisitos; (II) Classificação e Organização de Requisitos; (IV) Priorização e Negociação de Requisitos; e (I) Documentação de Requisitos.

**Gabarito:** Letra C

**11. (FCC / SEMEF-AM – 2019)** O processo de validação de requisitos de software deve ser utilizado em um projeto da Fazenda Municipal, sendo que seus técnicos de TI, devem, nesse processo de validação, efetuar revisões de requisitos, atentando que a propriedade:

- a) facilidade de compreensão analisa se o requisito pode ser excluído sem prejuízo ao sistema.
- b) adaptabilidade verifica se o requisito pode ser alterado sem afetar, de forma significativa, os demais requisitos.
- c) rastreabilidade verifica se o requisito pode ser testado, de forma completa.
- d) facilidade de verificação examina se requisito pode ser excluído sem prejuízo ao sistema.
- e) facilidade de compreensão analisa se o requisito tem sua origem diretamente estabelecida.

**Comentários:**

ATRIBUTOS	DESCRIÇÃO
VALIDADE	Examina se as partes interessadas que contribuíram com o levantamento de requisitos aceitam a especificação final obtida.
CONSISTÊNCIA	Examina se existem conflitos entre os requisitos identificados.
COMPREENSIBILIDADE	Examina se os requisitos são compreendidos de forma inequívoca pelas partes interessadas.
COMPLETUDE	Examina se todas as funcionalidades pretendidas fazem parte da especificação do sistema.
REALISMO	Examina se, dadas as restrições do projeto (tecnológicas, financeiras e temporais), o sistema especificado é implementável.
VERIFICABILIDADE	Examina se os requisitos foram descritos de forma que seja possível verificar se foram ou não implementados.
RASTREABILIDADE	Examina se a origem de cada requisito está claramente identificada.
ADAPTABILIDADE	Examina se os requisitos podem sofrer alterações sem produzir efeitos em outros requisitos.
CONFORMIDADE COM NORMAS	Examina se a especificação obedece às normas técnicas utilizadas para o desenvolvimento do sistema.

(a) Errado, examina se os requisitos são compreendidos de forma inequívoca pelas partes interessadas; (b) Correto, está perfeita a definição da propriedade; (c) Errado, examina se a origem



de cada requisito está claramente identificada; (d) Errado, examina se os requisitos foram descritos de forma que seja possível verificar se foram ou não implementados; (e) Errado, essa é a propriedade da rastreabilidade.

**Gabarito:** Letra B

**12. (FCC / Prefeitura de Manaus-AM – 2019)** Considerando a análise de requisitos, as informações de rastreabilidade desempenham papel de grande importância. Assim, a equipe responsável da Fazenda Municipal deve estar ciente de que a rastreabilidade de projeto significa:

- a) definir o mapeamento entre os requisitos de projeto e os usuários do sistema.
- b) listar os compiladores utilizados no desenvolvimento de cada módulo de software.
- c) determinar o mapeamento entre os requisitos de projeto e os locais onde o sistema será utilizado.
- d) determinar o desempenho de cada um dos requisitos do sistema.
- e) possuir o mapeamento entre os requisitos e os módulos de projeto que implementam os requisitos.

#### Comentários:

A rastreabilidade é a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados. Ela é frequentemente representada por meio de matrizes de rastreabilidade que mapeiam os requisitos aos stakeholders, aos outros requisitos, aos módulos de projeto, aos artefatos ou subprodutos, entre outros. Logo, a rastreabilidade de projeto significa possuir o mapeamento entre os requisitos e os módulos de projeto que implementam os requisitos.

**Gabarito:** Letra E

**13. (FCC / SEFAZ-SC – 2018)** A definição de contextos para que os usuários possam agir de maneira semelhante, entendendo melhor quais informações precisam fornecer durante a atividade de elicitação de requisitos, pode ser obtida por meio da aplicação de duas técnicas de elicitação denominadas:

- a) cenários e protótipos.
- b) entrevistas e observação.
- c) protótipos e observação.
- d) cenários e histórias de usuários.
- e) reuniões com facilitadores e histórias de usuários.

#### Comentários:



A questão deseja saber duas técnicas que definem contextos para que os usuários possam agir de maneira semelhante, isto é, que definem ambientes próximos ao da realidade. Com esse foco, as técnicas que mais "simulam" a realidade são: cenários e protótipos.

**Gabarito:** Letra A

**14. (FCC / SEFAZ-SC – 2018)** Durante o processo de validação, diferentes tipos de verificação podem ser efetuados com os requisitos registrados nos documentos de requisitos. O tipo de verificações de consistência é realizado para:

- a) identificar, por meio de análise mais aprofundada, outras funções necessárias, adicionais ou diferentes, além daquelas que um usuário pensava que fossem as necessárias para o sistema executar determinadas funções.
- b) evitar que requisitos, no documento, entrem em conflito uns com outros, ou seja, não deve haver restrições contraditórias ou descrições diferentes para mesma função do sistema.
- c) garantir que o documento de requisitos contenha os requisitos que definem todas as funções e as restrições pretendidas pelos usuários do sistema.
- d) assegurar, usando o conhecimento das tecnologias existentes, que os requisitos verificados possam ser realmente implementados, considerando o orçamento e o cronograma para o desenvolvimento do sistema.
- e) reduzir o potencial de conflito entre o cliente e o contratante por meio de um conjunto de testes que demonstre que o sistema entregue atende a cada requisito especificado.

#### Comentários:

- (a) Errado, trata-se de verificação de validade;
- (b) Correto, trata-se de verificação de consistência;
- (c) Errado, trata-se de verificação de completude;
- (d) Errado, trata-se de verificação de realismo;
- (e) Errado, trata-se da facilidade de verificação.

**Gabarito:** Letra B

**15. (FCC / SABESP – 2018)** Um Analista necessita levantar os requisitos de um sistema junto aos usuários. São técnicas de levantamento:

- a) Cenários e Peer Review.
- b) Product Owner e Brainstorming.
- c) Overview e Use Cases.



- d) Joint Application Design (ou Development) – JAD e Etnografia.
- e) Prototipação e Sprint.

#### Comentários:

- (a) Errado, Peer Review não é uma técnica de levantamento de requisitos;
- (b) Errado, Product Owner não é uma técnica de levantamento de requisitos;
- (c) Errado, Overview não é uma técnica de levantamento de requisitos;
- (d) Correto, JAD e Etnografia são ambas técnicas de levantamento de requisitos;
- (e) Errado, Sprint não é uma técnica de levantamento de requisitos.

**Gabarito:** Letra D

---

**16.(FCC / FUB – 2018)** O documento de requisitos deve ser elaborado a partir da análise de viabilidade do software, seguida de análise, especificação e validação de requisitos.

#### Comentários:

O processo de Engenharia de Requisitos inclui quatro atividades de alto nível: Estudo de Viabilidade; Elicitação e Análise; Especificação e Validação. Logo, a ordem apresentada na questão está perfeita!

**Gabarito:** Correto

---

**17.(FCC / BNB – 2018)** A revisão técnica é um procedimento utilizado para validar os requisitos de um projeto, com o objetivo de identificar eventuais inconsistências e verificar se os artefatos estão de acordo com o padrão esperado.

#### Comentários:

Em uma revisão formal de requisitos, a equipe de desenvolvimento conduz o cliente pelos requisitos de sistema, explicando as implicações de cada requisito. A equipe de revisão verifica cada requisito de acordo com diversos critérios para evitar conflitos, contradições, erros e omissões nos requisitos apontados e registrados formalmente no relatório de revisão. Logo, a revisão técnica é realmente um procedimento utilizado para validar os requisitos de um projeto, com o objetivo de identificar eventuais inconsistências e verificar se os artefatos estão de acordo com o padrão esperado.

**Gabarito:** Correto

---

**18.(FCC / DPE-AM – 2018)** Considere, por hipótese, que uma equipe de Analistas de Sistemas da Defensoria elencou a lista de requisitos para um novo sistema:





- O sistema não deverá revelar aos usuários nenhuma informação pessoal sobre os cidadãos, além do número do processo, em respeito à legislação de privacidade.
- Em razão das restrições referentes aos direitos autorais, alguns documentos devem ser excluídos imediatamente ao serem fornecidos pelos cidadãos em seus processos.
- O sistema deve implementar interfaces utilizando as normas de usabilidade vigentes para o serviço público.

A lista apresenta exemplos de requisitos:

- a) funcionais do tipo proteção e do tipo regulação.
- b) funcionais de usabilidade.
- c) não-funcionais de proteção.
- d) funcionais internos de legislação.
- e) não-funcionais externos do tipo legal e do tipo regulador.

#### Comentários:

O primeiro é um **RNF Externo** do tipo **Legal**; o segundo é um **RNF Externo** do tipo **Legal**; e o terceiro é um **RNF Externo** do tipo **Regulador**.

**Gabarito:** Letra E

**19.(FCC / BNB – 2018)** No levantamento de informações, os requisitos dos solicitantes são classificados como normais e conceituais. Os requisitos normais refletem os objetivos e as metas do produto, ao passo que os conceituais estão implícitos no produto ou extrapolam as expectativas do cliente.

#### Comentários:

No levantamento de informações, os requisitos dos solicitantes são classificados como normais e conceituais, **esperados e fascinantes**. Os requisitos normais refletem os objetivos e as metas do produto, ao passo que os conceituais **esperados** estão implícitos no produto **ou e os requisitos fascinante** extrapolam as expectativas do cliente.

**Gabarito:** Errado

**20.(FCC / BNB – 2018)** O protótipo operacional serve para aprimorar o entendimento de como o sistema deve funcionar, por meio da elucidação dos requisitos do usuário e da compreensão de suas necessidades.

#### Comentários:



Perfeito! O protótipo operacional realmente serve para aprimorar o entendimento de como o sistema deve funcionar, por meio da elucidação dos requisitos do usuário e da compreensão de suas necessidades. Ele pode ser utilizado tanto na elicitacão quanto na validacão de requisitos.

**Gabarito:** Correto

**21. (FCC / MPE-MA – 2013)** O escopo de um projeto é determinado pelo levantamento de requisitos funcionais e não funcionais. Dentre os requisitos não funcionais se enquadram os requisitos organizacionais, que podem ser divididos em:

- a) reguladores e éticos.
- b) ambientais, operacionais e de desenvolvimento.
- c) contábeis e de segurança.
- d) de desempenho e de espaço.
- e) de eficiência, de confiança e de proteçãõ.

#### Comentários:

Os Requisitos Organizacionais podem ser divididos em Requisitos Ambientais, Operacionais e de Desenvolvimento (ou Implementacão).

**Gabarito:** Letra B

**22. (FCC / DPE-SP – 2013)** Em uma das etapas da Engenharia de Requisitos há a preocupacão em se observar a especificacão produzida, visando verificar que os requisitos tenham sido declarados, por exemplo, sem ambiguidades.

O texto refere-se à etapa de:

- a) gestãõ dos requisitos.
- b) elicitacão dos requisitos.
- c) negociaçãõ dos requisitos.
- d) levantamento dos requisitos.
- e) validacão dos requisitos.

#### Comentários:

Galera, os produtos de trabalho resultantes da engenharia de requisitos são avaliados quanto a qualidade no processo de validacão. A validacão dos requisitos examina a especificacão para garantir que todos os requisitos do software tenham sido declarados de modo não ambíguo.

**Gabarito:** Letra E



**23. (FCC / TST – 2012)** Na Engenharia de Requisitos, o gerente de requisitos:

- a) acompanha e monitora ações durante a verificação do software, sendo este o processo que garante o atendimento aos requisitos informados pelo usuário final.
- b) possui autonomia para realizar alterações no projeto para garantir que o software seja bem construído e atenda às necessidades da equipe de desenvolvimento.
- c) mantém atualizados os requisitos junto ao usuário final e a equipe de desenvolvimento, a fim de obter sucesso no processo de homologação do software, atendendo as necessidades e expectativas.
- d) classifica os requisitos em diferentes tipos, sendo os do tipo funcional relacionados com o custo e confiabilidade do software e os do tipo não-funcional relacionados com os casos de uso.
- e) obtém o comprometimento dos integrantes da equipe de desenvolvimento de software para o cumprimento do processo de software.

#### Comentários:

- (a) Errado, ele trata da validação e, não, da verificação; (b) Errado, para que atenda às necessidades do usuário e, não, da equipe de desenvolvimento; (c) Correto, essa é uma de suas responsabilidades! (d) Errado, a questão inverteu os conceitos; (e) Errado, essa não é uma função do Gerente de Requisitos.

**Gabarito:** Letra C

**24. (FCC / TCE-AP – 2012)** Em relação a requisitos de sistemas, considere:

- I. O modo como um sistema deve reagir a certas entradas e o comportamento em que o sistema deve ter em certas situações e, em alguns casos, especificar o que o sistema não deve fazer, são chamados de requisitos não-funcionais.
- II. As restrições aos serviços ou funções de um sistema, como, por exemplo, processos de desenvolvimento ou utilização de padrões, são requisitos de funcionamento do sistema ou requisitos funcionais.
- III. Requisitos que vem do domínio da aplicação do sistema e refletem características ou restrições para aquele domínio são chamados de requisitos de domínio e podem ser requisitos funcionais e/ou não-funcionais.

Está correto o que se afirma em:



- a) III, apenas.
- b) I, II e III.
- c) I e II, apenas.
- d) II e III, apenas.
- e) I, apenas.

#### Comentários:

(a) Errado, esses são chamados de Requisitos Funcionais; (b) Errado, esses são Requisitos Não-Funcionais; (c) Correto, esses são os chamados Requisitos de Domínio.

**Gabarito:** Letra A

**25. (FCC / MPE-PE – 2012)** Os requisitos não funcionais não estão diretamente ligados aos serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área, entre outros. Dentre os tipos de requisitos não funcionais, é possível destacar os requisitos de produto, organizacionais e externos. Dentre os requisitos de produto, podemos citar os requisitos:

- a) de eficiência e de confiança.
- b) contábeis e de desempenho.
- c) legais e de usabilidade.
- d) reguladores e de proteção.
- e) legais e contábeis.

#### Comentários:

Os Requisitos de Produto são: Desempenho, **Confiabilidade**, Portabilidade, **Eficiência**, Performance e Espaço (8ª Edição); ou Usabilidade, **Eficiência**, Desempenho, Espaço, Proteção e **Confiabilidade** (9ª Edição).

**Gabarito:** Letra A

**26. (FCC / TRE-CE – 2012)** Considere:

- I. Para cada cliente deve ser aplicado um identificador único.
- II. O tempo de resposta entre a requisição e a informação não pode exceder a 2 ms.
- III. Clientes têm filiais que devem "carregar", na base de dados, o identificador do cliente principal.



IV. O sistema não deve ferir as leis de proteção ambiental.

São requisitos não funcionais os que constam em

- a) I e II, apenas.
- b) II e III, apenas.
- c) II e IV, apenas.
- d) I, III e IV, apenas.
- e) I, II, III e IV.

### Comentários:

Galera, essa questão é bastante polêmica. *Por que?* Eu acho que é bem claro que os itens II (que fala de desempenho) e IV (que fala de leis externas) são requisitos não-funcionais. O problema são os itens I e III – o primeiro, na minha opinião, também é claramente um RNF e o único que pode gerar alguma dúvida é o último, que eu também considero como RNF. Ou seja, para mim, todos são RNF, mas a banca considerou que apenas os itens II e IV são RNF – discordo veementemente!

**Gabarito:** Letra C

**27. (FCC / TST – 2012)** Na Engenharia de Requisitos, o gerente de requisitos:

- a) acompanha e monitora ações durante a verificação do software, sendo este o processo que garante o atendimento aos requisitos informados pelo usuário final.
- b) possui autonomia para realizar alterações no projeto para garantir que o software seja bem construído e atenda às necessidades da equipe de desenvolvimento.
- c) mantém atualizados os requisitos junto ao usuário final e a equipe de desenvolvimento, a fim de obter sucesso no processo de homologação do software, atendendo as necessidades e expectativas.
- d) classifica os requisitos em diferentes tipos, sendo os do tipo funcional relacionados com o custo e confiabilidade do software e os do tipo não-funcional relacionados com os casos de uso.
- e) obtém o comprometimento dos integrantes da equipe de desenvolvimento de software para o cumprimento do processo de software.

### Comentários:

(a) Errado. Não é durante a Verificação, mas Validação de Software; (b) Errado. Não são as necessidades da Equipe de Desenvolvimento, mas dos usuários; (c) Correto. É exatamente isso; (d)



Errado. Estão invertidos, custo e confiabilidade são RNF e Casos de Uso são RF; (e) Errado. Isso é responsabilidade do Gerente de Projetos e, não, do Gerente de Requisitos.

**Gabarito:** Letra C

**28.(FCC / TJ-PE – 2012)** Na engenharia de requisitos trata-se de uma técnica de elicitação que ocorre em ambiente mais informal em que toda a idéia deve ser levada em consideração para a solução de um problema, sendo proibida a crítica a qualquer sugestão dada, e encorajada, inclusive, a criação de ideias que pareçam estranhas ou exóticas:

- a) Prototipação.
- b) Entrevista.
- c) Questionário.
- d) Brainstorming.
- e) Análise de protocolos.

**Comentários:**

A questão trata da Técnica de *Brainstorming*, isto é, uma abordagem de elicitação ocorrida em grupo em ambientes informais durante cerca de quinze minutos em que toda a ideia deve ser levada em consideração, sendo proibida a crítica a qualquer sugestão dada, e encorajada, inclusive, a criação de ideias que pareçam estranhas ou exóticas.

**Gabarito:** Letra D

**29.(FCC / INFRAERO – 2011 – Letra D)** No contexto de levantamento de requisitos, funcionalidade é um dos aspectos que deve ser levado em conta na abordagem dos requisitos funcionais.

**Comentários:**

Ué, claro! A funcionalidade é evidentemente um dos aspectos na abordagem dos requisitos funcionais.

**Gabarito:** Correto

**30.(FCC / INFRAERO – 2011)** A engenharia de requisitos ajuda os engenheiros de software a compreender melhor o problema que eles vão trabalhar para resolver. Ela inclui um conjunto de tarefas que levam a um entendimento de qual será o impacto do software sobre o negócio, do que o cliente quer e de como os usuários finais vão interagir com o software. A função de negociação no processo de engenharia de requisitos:

- a) especifica, revisa e valida o problema de modo a garantir que seu entendimento e o entendimento do cliente sobre o problema coincidam.





- b) refina e modifica os requisitos. É uma ação de modelagem de análise composta de várias tarefas de modelagem e refinamento.
- c) define quais são as prioridades, o que é essencial, o que é necessário. Clientes, usuários e outros interessados são solicitados a ordenar os requisitos e depois discutir os conflitos de prioridade.
- d) ajuda o cliente a definir o que é necessário.
- e) define o escopo e a natureza do problema a ser resolvido.

### Comentários:

Galera... falou em Negociação, tem que lembrar de consenso entre as partes interessadas. E para chegar a um consenso, deve-se definir prioridades. Na atividade de priorização e negociação, o engenheiro de requisitos deve conciliar os conflitos por meio de uma negociação entre clientes, usuários e partes interessadas. Pergunta-se a eles quais são seus requisitos prioritários. Então, utiliza-se uma abordagem iterativa que avalia custos e riscos de modo a balancear todas as demandas.

**Gabarito:** Letra C

**31. (FCC / TRT1 – 2011)** A técnica utilizada na compreensão de requisitos sociais e organizacionais por observação das rotinas dos envolvidos é a:

- a) prototipação.
- b) por pontos de vista.
- c) por cenário.
- d) entrevista.
- e) etnografia.

### Comentários:

*Observação de Rotinas?* Trata-se de Etnografia – técnica de observação utilizada para compreender os requisitos organizacionais e sociais. Coloca-se o analista dentro do campo de atuação dos usuários, observando o trabalho diário anotando as tarefas reais em que os participantes estão envolvidos.

**Gabarito:** Letra E



**32. (FCC / INFRAERO – 2011)** Os produtos de trabalho resultantes da engenharia de requisitos são avaliados quanto à qualidade durante a etapa de validação de requisitos. Analise os itens a seguir referentes a essa etapa:

- I. Um dos principais mecanismos de validação de requisitos é a avaliação técnica formal.
- II. O modelo de análise pode garantir que os requisitos foram consistentemente declarados.
- III. É frequentemente útil examinar cada requisito em face de um conjunto de questões do tipo checklist.
- IV. A equipe de revisão que avalia os requisitos inclui apenas pessoas com conhecimento técnico na área de TI, como engenheiros de softwares, desenvolvedores etc.

Está correto o que consta em:

- a) I, II, III e IV.
- b) II e IV, apenas.
- c) I, II e IV, apenas.
- d) II, III e IV, apenas.
- e) I, II e III, apenas.

#### Comentários:

(I) Correto! Um dos principais mecanismos de validação de requisitos é a avaliação/revisão técnica formal; (II) Correto! Como eu sempre falo, quando uma questão usa “pode”, ela só estará errada se não existir nenhuma outra hipótese. Ora, não há nenhuma hipótese de um modelo de análise garantir que os requisitos foram consistentemente declarados? Não, é possível, logo questão perfeita; (III) Correto! Embora a revisão para validação dos requisitos possa ser conduzida de qualquer maneira desde que possibilite a descoberta de erros nos requisitos, é útil examinar cada requisito contra um *checklist*. (IV) O time de revisão inclui os engenheiros de sistema, clientes, usuários e outros stakeholders que examinam a especificação do sistema à procura de erros de conteúdo ou interpretação, pontos onde pode ser necessário esclarecimento, perda de informações, inconsistências (um dos maiores problemas da engenharia de grandes produtos), requisitos conflitantes, ou requisitos irrealistas (de desenvolvimento impossível).

**Gabarito:** Letra E

**33. (FCC / TRT19 – 2011)** A avaliação do impacto de mudança de um requisito, muitas vezes, faz com que seja necessário retornar à sua fonte. Na validação dos requisitos, a equipe deve estar atenta, portanto, à:

- a) rastreabilidade.



- b) adaptabilidade.
- c) qualidade.
- d) facilidade de compreensão.
- e) facilidade de verificação.

### Comentários:

Trata-se da Rastreabilidade, isto é, avaliar impacto no projeto da mudança em um requisito. É a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados. Ela é frequentemente representada por meio de matrizes de rastreabilidade que relacionam os requisitos aos stakeholders, aos outros requisitos, aos módulos de projeto, aos artefatos ou subprodutos, etc.

**Gabarito:** Letra A

**34. (FCC / TRT23 – 2011)** Tabelas de rastreamento para relacionar os requisitos identificados a um ou mais aspectos do sistema ou do seu ambiente devem ser desenvolvidas, segundo Pressman, na engenharia de requisitos por meio da função de:

- a) gestão.
- b) especificação.
- c) elaboração.
- d) negociação.
- e) validação.

### Comentários:

De acordo com Pressman, a gestão de requisitos começa com a identificação. A cada requisito é atribuído um modo identificador. Uma vez identificados os requisitos, tabelas de rastreamento são desenvolvidas. Lembrando que a rastreabilidade é a propriedade de uma especificação de requisitos que reflete a facilidade de encontrar os requisitos relacionados

**Gabarito:** Letra A

**35. (FCC / BAHIA GÁS – 2010)** É uma restrição sobre os serviços ou as funções oferecidos pelo sistema. Pode ser uma restrição de timing, sobre o processo de desenvolvimento, sobre o desempenho ou sobre a confiabilidade do sistema, entre outras. Trata-se de:

- a) requisito não funcional.
- b) requisito funcional.
- c) especificação de risco.
- d) iteração de processo.
- e) etnografia.



### Comentários:

A questão trata de Requisito Não-Funcional, isto é, são restrições nos serviços ou funções oferecidas pelo sistema.

**Gabarito:** Letra A

**36.(FCC / DPE-SP – 2010)** Sobre análise de requisitos da engenharia de software, considere:

- I. Os requisitos de usuário podem descrever tanto requisitos funcionais quanto requisitos não-funcionais.
- II. Os requisitos de sistema podem descrever apenas requisitos não funcionais.
- III. Os requisitos não-funcionais podem ser divididos em requisitos de produto, organizacionais e externos.

Está correto o que se afirma em:

- a) III, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

### Comentários:

(a) Correto! É exatamente isso. Lembrando que Requisitos de Usuários são descrições, em linguagem natural e com diagramas, de quais serviços o sistema deve fornecer e as restrições sob as quais deve operar. São requisitos com alto nível de abstração e poucos detalhes, feitos para serem lidos por pessoas leigas. (b) Errado! Requisitos de Sistema, de fato, podem ser funcionais ou não-funcionais. No entanto, o cerne dos requisitos de sistema são os requisitos funcionais. Não vejo a possibilidade de um documento de requisitos de sistema conterem apenas requisitos não-funcionais - o inverso talvez seja possível, isto é, um documento de requisitos de sistema com apenas requisitos funcionais. (c) Correto! Ele se divide nessas três categorias.

**Gabarito:** Letra C

**37.(FCC / DPE-SP – 2010)** No contexto da Engenharia de Requisitos, considere:

- I. O sistema deve fornecer uma entrada de dados que possibilite a inclusão de atributos de permissão de acesso às dependências da corporação por técnicos, supervisores e chefes.



II. Algumas permissões de acesso deverão ter tratamento especial para a entrada de atributos. Para este tipo de permissão, atributos excedentes a uma faixa predeterminada só poderão ser incluídos por chefes de seção.

Em relação às assertivas acima, é correto afirmar:

- a) O item I trata de um requisito funcional e a ele está associado o requisito não funcional, contido no item II.
- b) O item I trata de um requisito não funcional e a ele está associado o requisito funcional, contido no item II.
- c) Ambos referem-se a requisitos funcionais.
- d) A assertiva contida no item II é uma condição restritiva do requisito não funcional do item I. Por si só, não constitui um requisito, tanto funcional quanto não funcional.
- e) A assertiva contida no item II é uma condição restritiva do requisito funcional do item I. Por si só, não constitui um requisito, tanto funcional quanto não funcional.

### Comentários:

Galera, o primeiro item trata de um possível serviço do sistema, logo é um Requisito Funcional. Já o segundo item, da maneira que está escrito, trata-se de uma Regra de Negócio, isto é, não é sequer um Requisito. *É possível transformá-lo em um Requisito? Sim, percebam:*

*II. O Sistema deve permitir um tratamento especial para indivíduos com determinada permissão de acesso, isto é, atributos excedentes só poderão ser incluídos por chefes de sessão.*

Pronto, nós transformamos a Regra de Negócio em um Requisito Funcional. Sommerville afirma que eles definem como o sistema deve reagir em condições e até o que o sistema não deve fazer (restrições). Portanto, acredito que a resposta é a Letra E, mas a banca entendeu que o segundo item é um Requisito Não-Funcional.

**Gabarito:** Letra A

**38.(FCC / MPE-RN – 2010)** Na engenharia de software, etnografia é:

- a) uma fase do processo de software aplicada no modelo em cascata.
- b) uma fase do processo de software aplicada no modelo em espiral.



- c) uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais.
- d) uma técnica aplicada na engenharia de requisitos cujo objetivo é definir, a priori, as classes que contém elementos gráficos (BLOB).
- e) um projeto cujo principal objetivo é criar interfaces gráficas, que facilitam o acesso do usuário (GUI).

#### Comentários:

A técnica de Etnografia é uma Técnica de observação utilizada para compreender os requisitos organizacionais e sociais. Coloca-se o analista dentro do campo de atuação dos usuários, observando o trabalho diário anotando as tarefas reais em que os participantes estão envolvidos. Em geral, essa é uma técnica utilizada em conjunto com outras técnicas. Como ela é uma técnica de observação, isoladamente ela não é muito eficaz na elicitação.

**Gabarito:** Letra C

**39.(FCC / MPE-RN – 2010)** As políticas de rastreabilidade de requisitos são decididas durante o estágio de:

- a) agregação dos requisitos funcionais, apenas.
- b) implementação do sistema, apenas.
- c) implementação do sistema
- d) eliminação dos requisitos não funcionais.
- e) gerenciamento de requisitos.

#### Comentários:

As políticas de rastreabilidade de requisitos são decididas durante o estágio de Gerenciamento de Requisitos.

**Gabarito:** Letra E

**40.(FCC / SEFAZ-SP – 2009)** *É necessário que o software calcule os salários dos diaristas e mensalistas e emita relatórios mensais sumariados por tipo de salário. Entretanto, a base de dados deve estar protegida e com acesso restrito aos usuários autorizados. De qualquer forma, o tempo de resposta das consultas não deve superar os quinze segundos, pois inviabilizaria todo o investimento nesse sistema. Devo lembrar que os relatórios individuais dos departamentos, nos quais constam os salários dos funcionários, devem ser emitidos quinzenalmente em razão dos adiantamentos e vales que recebem. É fundamental que o software seja operacionalizado usando código aberto. Necessito, ainda, forte gerenciamento de risco, prazo e custo, porque a entrega do*





*produto final não pode ultrapassar o prazo de oito meses a contar da data de início do projeto.* No texto, são requisitos funcionais:

- a) Calcule os salários dos diaristas e mensalistas e os relatórios individuais dos departamentos, nos quais constam os salários dos funcionários, devem ser emitidos quinzenalmente.
- b) Necessito, ainda, forte gerenciamento de risco, prazo e custo e a base de dados deve estar protegida e com acesso restrito aos usuários autorizados.
- c) É fundamental que o software seja operacionalizado usando código aberto e emita relatórios mensais sumariados por tipo de salário.
- d) Emita relatórios mensais sumariados por tipo de salário e necessito, ainda, forte gerenciamento de risco, prazo e custo.
- e) A base de dados deve estar protegida e com acesso restrito aos usuários autorizados e entrega do produto final não pode ultrapassar o prazo de oito meses.

### Comentários:

a) *Calcular os salários dos diaristas e mensalistas?* RF, visto que é uma possível funcionalidade que o sistema deve apresentar; *Emitir relatórios individuais quinzenalmente?* RF, visto que também é uma possível funcionalidade que o sistema deve apresentar.

b) *Gerenciamento de Risco, Prazo e Custo?* RNF, visto que é uma restrição do sistema; *Base de dados protegida e com acesso restrito?* RNF, visto que é uma restrição do sistema.

c) *Operacionalizar o software com código aberto?* RNF, visto que é uma restrição do sistema; *Emitir relatórios mensais sumarizados?* RF, é uma possível funcionalidade que o sistema deve apresentar.

d) *Emitir relatórios mensais sumarizados?* RF, visto que é uma possível funcionalidade que o sistema deve apresentar. *Gerenciamento de Risco, Prazo e Custo?* RNF, visto que é uma restrição do sistema.

e) *Base de dados protegida e com acesso restrito?* RNF, visto que é uma restrição do sistema; *Restrição de 8 meses quanto ao prazo?* RNF, visto que é uma restrição do sistema.

Observe que os Requisitos Funcionais estão mais relacionados a possíveis utilizações do usuário, em sua interação com o sistema e como o sistema vai realizar alguma atividade. Já os Requisitos Não Funcionais estão mais relacionados a aspectos qualitativos do sistema, de seu processo de desenvolvimento e de suas restrições. Logo, percebe-se que somente a Letra A possui ambos os Requisitos Funcionais!

**Gabarito:** Letra A



**41. (FCC / TRT3 – 2009)** Com relação aos requisitos de software, considere:

- I. funcionais são somente requisitos de usuário.
- II. funcionais e não-funcionais podem ser requisitos de usuário.
- III. funcionais e não-funcionais podem ser requisitos de sistema.

Está correto o que se afirma APENAS em

- a) I.
- b) II.
- c) III.
- d) I e III.
- e) II e III.

#### Comentários:

Requisitos funcionais e não-funcionais podem ser tanto requisitos de usuário quanto requisitos de sistema.

**Gabarito:** Letra E

**42. (FCC / TRT7 – 2009)** No processo de engenharia de requisitos, é uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais. Trata-se de:

- a) Workshop.
- b) Brainstorming.
- c) Scrum.
- d) Análise de ponto de vista.
- e) Etnografia.

#### Comentários:

*Já viram quantas vezes essa técnica caiu em provas?* De novo, Etnografia – técnica de observação utilizada para compreender os requisitos organizacionais e sociais.

**Gabarito:** Letra E

**43. (FCC / TRT3 – 2009)** São técnicas e abordagens utilizadas na obtenção dos requisitos:

- a) estresse, cenários e workshop.
- b) workshop, etnografia e estresse.
- c) etnografia, questionários e validação.



- d) pontos de vista, cenários e entrevista.
- e) pontos de vista, casos de uso e validação.

**Comentários:**

- (a) Errado, estresse não é uma técnica de obtenção de requisitos;
- (b) Errado, estresse não é uma técnica de obtenção de requisitos;
- (c) Errado, validação não é uma técnica de obtenção de requisitos;
- (d) Correto, todas são técnicas de obtenção de requisitos;
- (e) Errado, validação não é uma técnica de obtenção de requisitos;

---

**Gabarito:** Letra D

**44. (FCC / PGE-RJ – 2009)** No âmbito da Engenharia de Requisitos, uma revisão técnica formal é:

- a) um teste de desempenho.
- b) uma técnica de elicitação.
- c) um instrumento de rastreamento.
- d) o resultado do escopo.
- e) um mecanismo de validação.

**Comentários:**

A revisão técnica formal (revisão de requisitos) é uma técnica ou mecanismo de validação de requisitos – assim como a prototipação e a geração de casos de teste.

---

**Gabarito:** Letra E



## QUESTÕES COMENTADAS – FGV

1. (FGV / TJ-AP – 2024) A equipe de desenvolvimento de sistemas do TJAP está implementando uma aplicação de acordo com o Processo Unificado (PU). A equipe atribuiu cada disciplina do PU a um analista diferente. O analista José ficou responsável pela disciplina do PU que é iniciada no começo da fase de concepção, intensificada ao máximo na fase de elaboração e amenizada ao longo da fase de construção. José ficou responsável pela disciplina do PU de:
- a) teste;
  - b) análise;
  - c) projeto;
  - d) requisitos;
  - e) modelagem.

### Comentários:

(a) Errado. A disciplina de teste no Processo Unificado (PU) é fundamental para a verificação e validação do sistema. Ela começa a ser aplicada na fase de elaboração e ganha intensidade na fase de construção e transição, não correspondendo ao padrão descrito na pergunta;

(b) Errado. A disciplina de análise no PU está geralmente associada à compreensão dos requisitos e à especificação de casos de uso, sendo intensificada na fase de elaboração, mas não é mencionada como específica em cada fase como a disciplina de requisitos;

(c) Errado. A disciplina de projeto no PU se intensifica na fase de construção, quando a arquitetura e o design detalhado do sistema são desenvolvidos. Ela não é a principal responsável na fase de concepção;

(d) Correto. A disciplina de requisitos no PU é iniciada na fase de concepção para identificar e definir os requisitos do sistema, intensificada ao máximo na fase de elaboração, quando os requisitos são refinados e detalhados, e é amenizada na fase de construção, à medida que o foco se desloca para o desenvolvimento e implementação;

(e) Errado. A disciplina de modelagem no PU está relacionada ao desenvolvimento de diagramas e modelos para representar o sistema, sendo mais intensificada na fase de construção, quando o sistema é implementado, e não especificamente descrita como na fase de concepção e elaboração.

**Gabarito:** Letra D

2. (FGV / Câmara de Fortaleza – 2024) No contexto de requisitos de software, analise os itens a seguir.



- I. "O sistema deve cadastrar um usuário" é um tipo de Requisito Funcional.
- II. "O sistema deve ficar no ar em 99,9% do tempo" é um tipo de Requisito Funcional de Desempenho.
- III. "O sistema deve ser capaz de armazenar dados de 100 mil clientes" é um tipo de Requisito Não-Funcional de Escalabilidade.

Está correto o que se afirma em:

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

### Comentários:

- (a) Correto. Item I está correto ao definir um Requisito Funcional como a funcionalidade de cadastrar um usuário;
- (b) Errado. O item II está incorreto, pois "99,9% do tempo" é uma característica de disponibilidade, um Requisito Não-Funcional de Confiabilidade, e não um Requisito Funcional de Desempenho;
- (c) Correto. O item I está correto como Requisito Funcional e o item III está correto como Requisito Não-Funcional de Escalabilidade, que especifica a capacidade de armazenamento do sistema;
- (d) Errado. O item II está incorreto, e o item I, que é correto, não está incluído nesta alternativa;
- (e) Errado. O item II está incorreto, pois descreve um Requisito Não-Funcional de Confiabilidade, e não um Requisito Funcional de Desempenho.

**Gabarito:** Letra C

3. (FGV / SES-MT – 2024) Relacione as técnicas de elicitação e especificação de requisitos às suas respectivas definições:

- 1. Entrevistas
- 2. Etnografia
- 3. Histórias e Cenários
- 4. Casos de Uso

( ) Identifica os atores envolvidos, as funcionalidades principais, bem como a interação entre atores e funcionalidades do sistema



( ) Identifica características sobre o sistema (ou processo) atual e sobre o sistema que será desenvolvido a partir de questionamento aos stakeholders

( ) Descreve situação de uso do sistema, com informações como: ator(es); pré-condição; fluxo normal; fluxos alternativos; e pós-condição.

( ) Técnica de observação onde o analista (engenheiro de requisitos) se insere na organização do cliente, observa o trabalho no dia a dia e anota as tarefas dos funcionários

Assinale a opção que indica a relação correta, segundo a ordem apresentada.

a) 2, 4, 1 e 3.

b) 3, 2, 4 e 1.

c) 1, 3, 2 e 4.

d) 4, 1, 3 e 2.

### Comentários:

(4) Os casos de uso descrevem as interações entre usuários (atores) e o sistema, identificando funcionalidades e a forma de uso.

(1) As entrevistas são utilizadas para coletar informações sobre o sistema atual e o desejado diretamente dos stakeholders por meio de questionamentos.

(3) Histórias e cenários fornecem narrativas detalhadas de como o sistema será usado, incluindo atores e fluxo de eventos.

(2) A etnografia envolve a observação direta do trabalho e tarefas dentro da organização para entender o contexto e os processos existentes.

**Gabarito:** Letra D

4. (FGV / SES-MT – 2024) Com relação à Engenharia de Requisitos, analise as afirmativas a seguir e assinale (V) para a verdadeira e (F) para a falsa.

( ) A Engenharia de Requisitos inclui três atividades fundamentais: elicitacão (e análise); especificacão; e, validacão.

( ) Na etapa de classificacão e organizacão, no processo de elicitacão e análise de requisitos, o conjunto não estruturado de requisitos é agrupado pelos requisitos relacionados e organizado em grupos coerentes.

( ) Na prática, a Engenharia de Requisitos é um processo iterativo, no qual as atividades são intercaladas



As afirmativas são, respectivamente:

- a) V, F e F.
- b) F, V e F.
- c) F, F e V.
- d) V, V e V.

### Comentários:

(V) A Engenharia de Requisitos normalmente envolve essas três atividades principais para identificar, documentar e confirmar os requisitos do sistema;

(V) Esta descrição corresponde à prática comum de classificação e organização de requisitos durante a fase de análise para facilitar a gestão e compreensão dos requisitos;

(V) A Engenharia de Requisitos é um processo iterativo, onde as atividades de elicitação, especificação e validação podem ser revisitadas várias vezes à medida que novos requisitos são descobertos ou modificados.

**Gabarito:** Letra D

5. (FGV / AL-TO – 2024) Sobre Engenharia de Requisitos, analise as afirmativas a seguir.

I. A validação de requisitos é um processo que visa assegurar que os requisitos atendem às necessidades e expectativas dos stakeholders.

II. As técnicas de elicitação de requisitos, como brainstorming, entrevistas e prototipagem, são utilizadas para coletar requisitos de forma passiva, sem interação com os stakeholders.

III. A priorização de requisitos ajuda a definir a ordem de importância para o projeto.

Está correto o que se afirma em:

- a) II, apenas.
- b) I e III, apenas.
- c) II e III, apenas.
- d) I e II, apenas.
- e) I, II e III.

### Comentários:

(I) Correto. A validação de requisitos é um processo essencial na Engenharia de Requisitos que visa verificar se os requisitos documentados realmente refletem as necessidades e expectativas dos





stakeholders. Isso assegura que o produto final estará alinhado com o que foi solicitado e esperado pelos usuários e clientes;

(II) Errado. As técnicas de elicitación de requisitos, como brainstorming, entrevistas e prototipagem, são altamente interativas e envolvem uma participação ativa dos stakeholders. Essas técnicas são utilizadas para coletar informações sobre os requisitos de forma dinâmica e envolvente, e não de maneira passiva;

(III) Correto. A priorização de requisitos é um componente fundamental na gestão de requisitos. Ela ajuda a equipe de projeto a entender quais requisitos são mais críticos e devem ser atendidos primeiro, otimizando recursos e alinhando o desenvolvimento às expectativas estratégicas do negócio.

**Gabarito:** Letra B

**6. (FGV / AL-TO – 2024)** A sociedade empresária Max Tech produz um aplicativo de gerenciamento de tarefas para equipes remotas. Após o lançamento, vários usuários relataram problemas de usabilidade e desempenho do aplicativo, especialmente em condições de rede variáveis. A diretoria reconheceu a necessidade de aprimorar a qualidade do software e solicitou uma revisão nos Requisitos Não Funcionais (RNFs) que foram negligenciados. No contexto apresentado, assinale a ação capaz de garantir a qualidade do software:

- a) Realizar uma pesquisa de mercado para identificar as principais funcionalidades ausentes no aplicativo e integrá-las no menor tempo possível.
- b) Priorizar a adição de novos recursos ao aplicativo, assumindo que isso automaticamente resolverá os problemas de desempenho e usabilidade relatados pelos usuários.
- c) Implementar uma estratégia de marketing mais agressiva para melhorar a percepção do produto pelo mercado, desviando a atenção dos problemas de desempenho e usabilidade.
- d) Focar na otimização do código para melhorar o desempenho do aplicativo, assim como aprimorar a interface com o usuário, garantindo uma execução mais rápida em todas as condições de rede e melhor usabilidade para o usuário.
- e) Conduzir uma revisão focada nos Requisitos Não Funcionais, especialmente aqueles relacionados à usabilidade e ao desempenho em condições de rede variáveis, para identificar lacunas e oportunidades de melhoria.

### Comentários:

(a) Errado. Embora realizar uma pesquisa de mercado para identificar funcionalidades ausentes possa ajudar a melhorar a atratividade do aplicativo, isso não garante a resolução dos problemas de usabilidade e desempenho já existentes;



- (b) Errado. A adição de novos recursos sem abordar os problemas fundamentais pode agravar os problemas de desempenho e usabilidade, em vez de resolvê-los;
- (c) Errado. Implementar uma estratégia de marketing mais agressiva pode melhorar temporariamente a percepção do produto, mas não resolve os problemas técnicos do aplicativo, podendo levar a uma maior insatisfação dos usuários a longo prazo;
- (d) Errado. Embora otimizar o código para melhorar o desempenho e aprimorar a interface do usuário sejam ações válidas, elas não abordam especificamente os Requisitos Não Funcionais que podem estar faltando ou mal definidos, como a capacidade de lidar com condições de rede variáveis;
- (e) Correto. Uma revisão focada nos Requisitos Não Funcionais, especialmente em aspectos relacionados à usabilidade e ao desempenho em condições de rede variáveis, é uma abordagem direcionada que pode identificar e corrigir as lacunas que causam os problemas relatados. Este processo ajuda a garantir que o software atenda às expectativas dos usuários em todos os aspectos técnicos relevantes.

**Gabarito:** Letra E

**7. (FGV / CGE-SC – 2023)** A respeito do requisito não funcional, assinale V para a afirmativa verdadeira e F para a falsa.

- ( ) O sistema deve permitir que os usuários criem e editem seus perfis.  
( ) O sistema deve estar disponível na Web e em dispositivos Android.  
( ) O sistema deve ter um recurso de pesquisa que permita aos usuários encontrar informações sobre vendas.  
( ) O sistema deve estar disponível 99,99% do tempo.  
( ) O sistema deve exibir dados de vendas em tempo real.

As afirmativas são, respectivamente,

- a) V – F – V – F – V.  
b) V – V – V – F – V.  
c) F – V – F – V – F.  
d) V – F – V – V – V.  
e) V – F – F – F – V.

### Comentários:

- (F) Trata-se de um Requisito Funcional;  
(V) Trata-se de um Requisito Não-Funcional;  
(F) Trata-se de um Requisito Funcional;



- (V) Trata-se de um Requisito Não-Funcional;
- (V) Trata-se de um Requisito Não-Funcional;

A questão considerou que “o sistema deve exibir dados de vendas em tempo real” é um Requisito Funcional. Eu discordo e acredito que a questão caberia recurso.

**Gabarito:** Letra C

**8. (FGV / AL-MA – 2023)** A elicitação de requisitos é uma atividade central da engenharia de requisitos. Assinale a técnica de elicitação que extrai informações necessárias para construir um novo sistema a partir do código fonte de um sistema legado:

- a) arqueologia de sistemas.
- b) entrevista dos stakeholders.
- c) leitura baseada em perspectiva.
- d) mapas mentais.
- e) observação de campo.

#### Comentários:

(a) Correto. Arqueologia de sistemas é a técnica de elicitação que extrai informações necessárias para construir um novo sistema a partir do código-fonte de um sistema legado. Essa abordagem envolve analisar o código existente para entender a funcionalidade, lógica de negócio, e requisitos que podem não estar documentados ou compreendidos por stakeholders atuais.

(b) Errado. Entrevista dos stakeholders é uma técnica de elicitação que envolve conversar diretamente com as partes interessadas para obter informações sobre os requisitos do sistema, mas não se concentra em extrair informações do código-fonte.

(c) Errado. Leitura baseada em perspectiva é uma técnica usada para revisar documentos ou código considerando diferentes perspectivas (como a de um desenvolvedor ou usuário), mas não é especificamente focada na extração de requisitos de sistemas legados.

(d) Errado. Mapas mentais são diagramas que organizam informações de maneira visual para ajudar na compreensão e brainstorming de ideias, mas não são usados para extrair informações de código-fonte de sistemas legados.

(e) Errado. Observação de campo envolve observar o uso do sistema em seu ambiente real para entender como ele funciona, mas não lida com a análise direta do código-fonte para extrair requisitos.

**Gabarito:** Letra A



9. (FGV / TCE-SP – 2023) A Equipe de Sistemas (ES) de um tribunal está trabalhando na especificação de requisitos para o Sistema de Protocolos (SisProt). O representante do cliente solicitou que o SisProt funcione 24 horas por dia, durante todo o ano, e explicou que será tolerada a interrupção do serviço prestado pelo SisProt por, no máximo, 1 hora por mês, seja para manutenções, seja por motivos extra sistema. O requisito solicitado pelo representante do cliente é categorizado como um requisito de:

- a) instabilidade;
- b) confiabilidade;
- c) manutenibilidade;
- d) tempo de resposta;
- e) responsabilidade (“accountability”).

#### Comentários:

(a) Errado. Instabilidade se refere a frequentes mudanças nos requisitos ou comportamentos do sistema, o que não é o caso aqui;

(b) Correto. O requisito de funcionar 24 horas por dia, com uma interrupção máxima tolerada de 1 hora por mês, é um requisito de confiabilidade, pois trata da disponibilidade e continuidade do serviço;

(c) Errado. Manutenibilidade se refere à facilidade com que o sistema pode ser mantido e atualizado, mas não é diretamente sobre a disponibilidade do serviço;

(d) Errado. Tempo de resposta se refere à rapidez com que o sistema responde às solicitações dos usuários, não à disponibilidade ou tempo de operação;

(e) Errado. Responsabilidade (“accountability”) se refere à capacidade de rastrear e responsabilizar usuários ou sistemas por suas ações, não à disponibilidade do serviço.

**Gabarito:** Letra B

10. (FGV / TCE-SP – 2023) O sistema SisBRAVO foi desenvolvido aderente ao preconizado na Lei nº 13.709/2018 – Lei Geral de Proteção de Dados (LGPD). O SisBRAVO solicita autorização para coleta de dados pessoais inseridos pelos usuários. Sendo assim, o SisBRAVO atende requisitos tipificados como:

- a) externos;
- b) organizacionais;
- c) de confiança;
- d) de confidencialidade;
- e) de adequação funcional.



### Comentários:

- (a) Correto. Requisitos externos referem-se a normas, leis e regulamentações que o sistema deve atender. A aderência à Lei Geral de Proteção de Dados (LGPD) é um requisito externo;
- (b) Errado. Requisitos organizacionais referem-se às necessidades e políticas internas da organização, não a leis e regulamentações externas;
- (c) Errado. Requisitos de confiança referem-se à confiabilidade e segurança do sistema em geral, mas não são específicos para regulamentações legais;
- (d) Errado. Requisitos de confidencialidade referem-se à proteção de informações contra acesso não autorizado, o que é parte da LGPD, mas a conformidade com a lei em si é um requisito externo;
- (e) Errado. Requisitos de adequação funcional referem-se às funcionalidades que o sistema deve fornecer para atender às necessidades dos usuários, mas não à conformidade legal.

**Gabarito:** Letra A

**11. (FGV / TCE-TO – 2022)** A Equipe de Desenvolvimento de Soluções (EDS) recebeu a solicitação de que um dos campos utilizados para entrada de dados da aplicação Web em construção apresente sugestões de palavras dinamicamente, conforme o usuário for digitando novos caracteres.

A EDS recebeu a solicitação de um requisito de:

- a) confiança;
- b) eficiência;
- c) desempenho;
- d) usabilidade;
- e) desenvolvimento.

### Comentários:

Trata-se claramente de um requisito de usabilidade, dado que as sugestões dinâmicas que surgem enquanto o usuário digita caracteres atuam diretamente na usabilidade no sentido de facilidade de uso do sistema.

**Gabarito:** Letra D

**12. (FGV / TCE-TO – 2022)** Carlos é uma parte interessada em uma aplicação Web e solicitou à equipe de desenvolvimento uma funcionalidade capaz de emitir relatórios com cabeçalhos padronizados. Assim, os cabeçalhos devem ter cor de fundo, paleta de cores e tipografia,

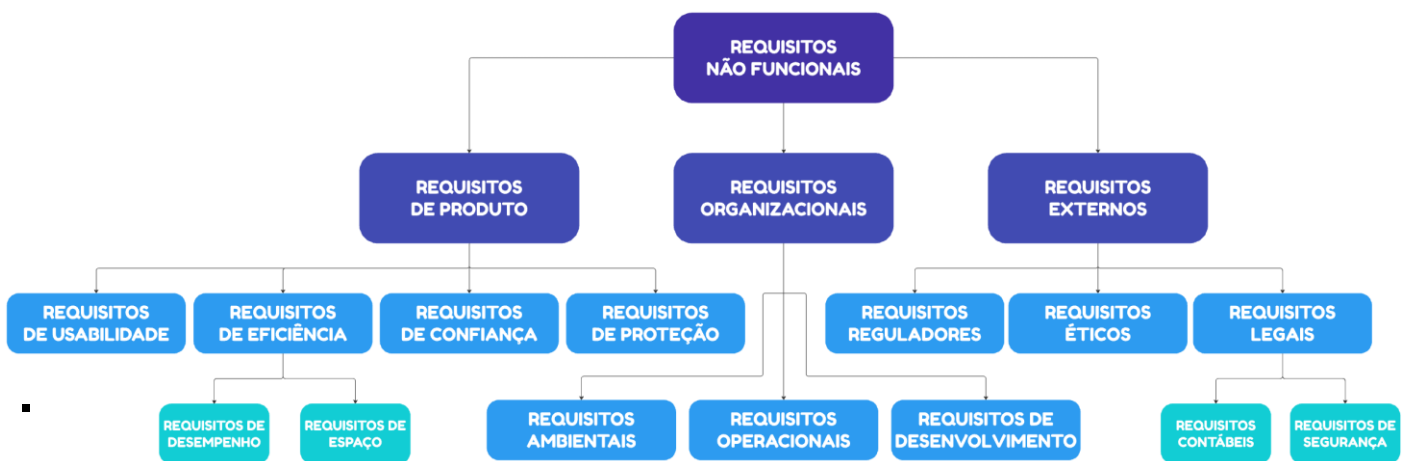


segundo o padrão adotado em outros documentos emitidos pelo departamento responsável pela aplicação.

A solicitação de Carlos refere-se a um requisito:

- a) funcional regulador;
- b) não funcional organizacional;
- c) funcional do processo operacional;
- d) não funcional de usabilidade;
- e) funcional de desenvolvimento.

### Comentários:



Padronização estética claramente não são requisitos funcionais, logo se trata de um requisito não funcional. Especificamente, trata-se de um requisito organizacional: **aqueles são derivados de políticas e procedimentos da organização do cliente e do desenvolvedor**. Entre os exemplos, estão padrões de processo que devem ser usados, linguagem de programação ou o método de projeto usado, e requisitos de entrega que especificam quando o produto e a sua documentação devem ser entregues.

**Gabarito:** Letra B

13. (FGV / TCE-TO – 2022) A Equipe de Desenvolvimento de Soluções (EDS) recebeu a solicitação de que um dos campos utilizados para entrada de dados da aplicação Web em construção apresente sugestões de palavras dinamicamente, conforme o usuário for digitando novos caracteres. A EDS recebeu a solicitação de um requisito de:

- a) confiança;
- b) eficiência;
- c) desempenho;
- d) usabilidade;
- e) desenvolvimento.



### Comentários:

Notem que se trata de uma solicitação que facilita a navegação intuitiva do usuário, logo se trata de um requisito de usabilidade.

**Gabarito:** Letra D

**14. (FGV / TCE-TO – 2022)** Carlos é uma parte interessada em uma aplicação Web e solicitou à equipe de desenvolvimento uma funcionalidade capaz de emitir relatórios com cabeçalhos padronizados. Assim, os cabeçalhos devem ter cor de fundo, paleta de cores e tipografia, seguindo o padrão adotado em outros documentos emitidos pelo departamento responsável pela aplicação:

A solicitação de Carlos refere-se a um requisito:

- a) funcional regulador;
- b) não funcional organizacional;
- c) funcional do processo operacional;
- d) não funcional de usabilidade;
- e) funcional de desenvolvimento.

### Comentários:

A solicitação de Carlos não tem nenhuma relação com a funcionalidade da aplicação web, logo se trata de um requisito não funcional. Essa solicitação tem o intuito de facilitar a navegação na aplicação web? Não, tem o intuito de padronizar os relatórios da organização, logo se trata de um requisito não funcional organizacional.

**Gabarito:** Letra B

**15. (FGV / TCE-TO – 2022)** As credenciais de acesso dos usuários de um aplicativo são armazenadas em um banco de dados e são utilizadas unicamente para acesso às funcionalidades do aplicativo. A equipe de desenvolvimento definiu como requisito não funcional que o sistema deve evitar que as senhas sejam obtidas por um invasor mesmo que o aplicativo ou banco de dados esteja comprometido.

Para implementar o requisito não funcional, um modo de proteger as senhas dos usuários é:

- a) ocultar o algoritmo utilizado para proteção das senhas;
- b) armazenar as senhas cifradas por meio de um algoritmo de chaves assimétricas;
- c) utilizar um algoritmo hash com salt antes de persistir as senhas no banco de dados;
- d) codificar um algoritmo próprio para cifrar as senhas com base em uma chave randômica segura;





e) usar uma chave randômica gerada pelo aplicativo para cifrar as senhas por meio de um algoritmo de chave simétrica.

### Comentários:

(a) Errado, o algoritmo pode ser público - é a chave que deve ser privada; (b) Errado, deve-se armazenar o hashing das senhas e, não, a senha cifrada por algoritmo assimétrico; (c) Correto, deve se utilizar um algoritmo de hash com salt\* para persistir os dados no banco; (d) Errado, codificar um algoritmo próprio é uma péssima ideia - o ideal é que o algoritmo seja pública para que sua segurança seja testada e validade; (e) Errado, não é um bom modo para proteger senhas de usuários.

\* Salt é uma string aleatória de dados adicionada à entrada de uma função hash cujo objetivo é dificultar a "pré-computação" de um conjunto de entradas que gerará uma determinada saída de hash. Como resultado, fica mais difícil para os invasores decifrar senhas ou outras informações confidenciais ao tentar adivinhar as entradas de uma função de hash.

Agora sendo rigoroso, eu acho que a questão poderia ser anulada. Todos esses são modos de proteger as senhas de usuário - alguns são péssimos, mas ainda assim são modos. Acredito que o enunciado da questão poderia questionar qual é a maneira mais adequada de proteger senhas de usuários.

**Gabarito:** Letra C

**16.(FGV / TCE-TO – 2022)** A Equipe de Tecnologia (ETi) de um tribunal de contas está levantando as necessidades para um novo sistema junto às partes interessadas. Uma das partes interessadas solicitou que o novo sistema seja fácil de usar, como requisito não funcional.

Para que o requisito não funcional "fácil de usar" seja objetivamente testado, a ETi deve considerar a métrica:

- a) eficiência;
- b) disponibilidade;
- c) tempo de treinamento;
- d) taxa de ocorrência de falhas;
- e) tempo de atualização de tela.

### Comentários:

O usuário deseja que o sistema seja fácil de usar, logo está relacionado ao tempo de treinamento. Essa métrica descreve quanto tempo é necessário para treinar um usuário para usar as funcionalidades de um sistema. Em geral, os tempos de treinamento estimados são baseados nas habilidades do usuário e no nível de complexidade do sistema.



**Gabarito:** Letra C

**17.(FGV / ALERJ – 2017)** O Analista de Sistemas Pedro está realizando um levantamento de requisitos por meio da prototipação. Sua intenção com esse protótipo é proporcionar uma visão geral do sistema com todas as suas funcionalidades, sem entrar em detalhes específicos de cada funcionalidade, de forma que a interface como um todo possa ser criticada pelos usuários. Nesse caso, o tipo de protótipo mais adequado é o (a):

- a) vertical;
- b) tridimensional;
- c) prototipação rápida;
- d) textual;
- e) horizontal.

### Comentários:

Pessoal, a prototipação horizontal prototipação horizontal consiste em montar uma interface completa em termos de elementos, permitindo uma visão geral do usuário por todo o sistema, assim como uma pessoa observa o horizonte, em um protótipo horizontal, o usuário obtém uma visão geral de tudo. A Prototipação Vertical é focada nas funcionalidades. Dessa forma, um protótipo do sistema pode ser reduzido a um módulo isolado, onde o usuário poderá ver em detalhes um pedaço limitado do sistema. A prototipação textual é aquela feita por meio da descrição (texto) das funcionalidades do sistema. Eu desconheço o conceito de Prototipação Rápida ocorre quando um software é construído rapidamente que apresenta a funcionalidade básico do produto desejado. O ponto-chave é que um protótipo rápido reflita a funcionalidade que o cliente vê, como telas de entrada e a geração de relatórios, mas omite aspectos não aparentes. Por fim, eu desconheço o conceito de prototipação tridimensional. Logo, nota-se que a questão trata da Prototipação Horizontal.

**Gabarito:** Letra E

**18.(FGV / BADESC – 2010)** Analise o fragmento a seguir:

"A base de dados deve ser protegida para acesso apenas de usuários autorizados".

O fragmento acima apresenta um exemplo do seguinte requisito:

- a) funcional.
- b) de usuário.
- c) de sistema.
- d) de domínio.
- e) não-funcional.



### Comentários:

A questão trata de um Requisito Não-Funcional! *Por que?* Porque isso não é uma funcionalidade a ser implementada por um programador e disponibilizada ao usuário. É, na verdade, uma restrição tecnológica exigida no sistema. Lembrando que os requisitos não funcionais estão raramente associados às características individuais do sistema. Trata-se de qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras.

**Gabarito:** Letra E

**19.(FGV / FIOCRUZ – 2010)** Sobre os processos de engenharia de requisitos, na elicitação e na análise ocorre total interação com os stakeholders no sistema, sendo o principal objetivo:

- a) a obtenção dos requisitos.
- b) a homologação do sistema.
- c) a elaboração do manual do usuário.
- d) a conversão de especificações em requisitos.
- e) a execução do estudo de viabilidade do sistema.

### Comentários:

Nessa atividade, os engenheiros de software trabalham com os clientes e os usuários finais do sistema para aprender sobre o domínio da aplicação, quais serviços o sistema deve fornecer, o desempenho esperado do sistema, restrições de hardware, etc. O objetivo principal dessa fase é a obtenção dos requisitos!

**Gabarito:** Letra A

**20.(FGV / MEC – 2009)** Requisitos não-funcionais estão diretamente relacionados com a satisfação dos usuários. Assinale a alternativa que não indique um requisito não-funcional:

- a) O sistema de arquivos deve ser protegido, para acesso, apenas, de usuários autorizados.
- b) O software deve ser implementado usando os conceitos de orientação a objetos.
- c) O tempo de desenvolvimento do software não deve ultrapassar seis meses.
- d) O software poderá ser executado em plataforma windows e linux.
- e) O software deve emitir relatórios de vendas a cada quinze dias.

### Comentários:

Todos, exceto o último, são requisitos não-funcionais – restrições sobre o sistema. Já o último é uma funcionalidade a ser oferecida pelo sistema.

**Gabarito:** Letra E



**21. (FGV / MEC – 2009)** As declarações de serviços que o sistema deve fornecer, de como ele deve reagir a entradas específicas ou se comportar em determinadas situações, são chamadas de requisitos:

- a) não-funcionais.
- b) de domínio.
- c) de sistema.
- d) funcionais.
- e) de usuário.

#### Comentários:

A questão trata dos Requisitos Funcionais, isto é, declarações de serviços que um sistema deve fornecer, como o sistema deve reagir a entradas específicas e como o sistema deve (ou não) se comportar em situações particulares.

**Gabarito:** Letra D

**22. (FGV / MEC – 2009)** Existem técnicas que são usadas na fase de levantamento de requisitos para coletar conhecimento dos usuários sobre os requisitos. Assinale a alternativa que indique apenas técnicas utilizadas na fase de levantamento de requisitos.

- a) JAD, WFMS, WBS, cenários e brainstorming.
- b) JAD, cenários, WFMS, questionários e intercepting.
- c) cenários, entrevistas, protótipos, workshop, brainstorming.
- d) leitura de documentos, protótipos, workshop, WBS e workflow.
- e) brainstorming, protótipos, workflow, leitura de documentos e intercepting.

#### Comentários:

A única opção que apresenta apenas técnicas de levantamento de requisitos é: Cenários, Entrevistas, Protótipos, Workshop e Brainstorm!

**Gabarito:** Letra C

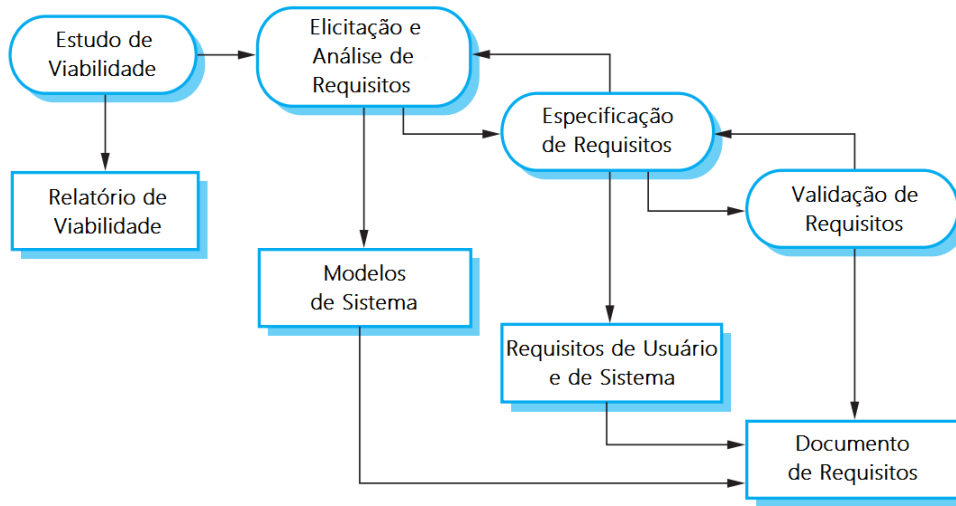
**23. FGV / Senado Federal – 2008)** Entre as atividades listadas a seguir, uma não faz parte da Engenharia de Requisitos. Assinale-a.

- a) estudo de viabilidade.
- b) análise de risco.
- c) levantamento de necessidades do cliente.
- d) verificação.



e) gerenciamento.

### Comentários:



Pessoal, questão complicada! Essas fases não existem exatamente dessa maneira em nenhuma bibliografia, então temos que fazer algumas associações. Estudo de Viabilidade é a primeira fase; Análise de Riscos não está em nenhum lugar; Levantamento de Necessidades do Cliente pode ser entendido como Elicitação de Requisitos; Verificação pode ser incluída na fase de Validação (apesar de eu não concordar); por fim, gerenciamento percorre de fato todas essas fases.

**Gabarito:** Letra B



## QUESTÕES COMENTADAS – DIVERSAS BANCAS

1. (CESGRANRIO / Caixa – 2024) Uma equipe de desenvolvimento ágil decidiu utilizar Pontos de História (Story Points) para estimar o esforço necessário para implementar as funcionalidades descritas no seu Product Backlog, na forma de Histórias do Usuário. Considerando-se esse contexto, a técnica de Pontos de História ajuda a equipe a compreender o trabalho envolvido em cada história por:
- a) alocar tarefas específicas da história do usuário aos membros da equipe.
  - b) classificar as histórias do usuário em categorias de prioridade, de alta a baixa.
  - c) permitir prever o tempo, em horas, que a equipe levará para completar uma história.
  - d) determinar o custo financeiro direto associado ao desenvolvimento de uma história do usuário.
  - e) estimar o esforço relativo necessário para implementar uma determinada história do usuário, comparando-o ao de outras histórias.

### Comentários:

- (a) Errado. Alocar tarefas específicas aos membros da equipe não é o objetivo dos Pontos de História;
- (b) Errado. A classificação de histórias em categorias de prioridade é feita através de técnicas de priorização, não pela atribuição de Pontos de História;
- (c) Errado. Pontos de História não se referem a tempo em horas, mas sim ao esforço relativo;
- (d) Errado. Pontos de História não determinam o custo financeiro direto do desenvolvimento;
- (e) Correto. Pontos de História são usados para estimar o esforço relativo necessário para implementar uma história do usuário, comparando-o ao esforço de outras histórias.

**Gabarito:** Letra E

2. (CESGRANRIO / IPEA – 2024) Um especialista em testes de software, com vasta experiência na criação de testes não funcionais, reuniu-se com sua equipe de desenvolvimento para avaliar se ainda havia requisitos não funcionais pendentes de teste em um novo software de e-commerce que estava sendo desenvolvido. Após ler a lista de requisitos identificados pela equipe como



pendentes de teste, o especialista identificou um requisito não funcional ainda não testado. Com base no cenário apresentado, o requisito não funcional identificado foi o de verificar se:

- a) a opção esqueci senha, ao ser pressionada pelo usuário, o direcionava para uma tela que solicitava o e-mail do usuário para recuperar tal senha.
- b) o tempo de resposta para aprovar ou não alguma solicitação era de até 8 segundos.
- c) o chatbot do sistema está apresentando o conteúdo esperado, a partir de informações fornecidas por algum usuário em um chat de ajuda acessado pelo software.
- d) algum dos cupons de desconto de 30% para certos produtos escolhidos estava sendo aplicado corretamente.
- e) os relatórios de venda do mês estavam sendo exportados no formato PDF e CSV.

### Comentários:

- (a) Errado. Verificar a funcionalidade da opção "esqueci a senha" é um teste funcional, pois avalia se uma funcionalidade específica está funcionando corretamente;
- (b) Correto. Verificar o tempo de resposta é um teste de desempenho, que é um tipo de teste não funcional. Esse teste avalia se o sistema atende aos requisitos de desempenho especificados;
- (c) Errado. Verificar se o chatbot apresenta o conteúdo esperado é um teste funcional, pois está avaliando o comportamento esperado de uma funcionalidade específica;
- (d) Errado. Verificar a aplicação correta dos cupons de desconto é um teste funcional, pois avalia se uma funcionalidade específica está funcionando conforme esperado;
- (e) Errado. Verificar a exportação de relatórios em formatos específicos é um teste funcional, pois está testando uma funcionalidade específica do sistema.

**Gabarito:** Letra B

3. (CESGRANRIO / IPEA – 2024) A engenharia de requisitos pode ser dividida em gerenciamento de requisitos e desenvolvimento de requisitos, e este, por sua vez, pode ser subdividido em elicitação, análise, especificação e validação. Cada um desses quatro componentes do desenvolvimento de requisitos possui um conjunto de boas práticas. Uma das boas práticas associadas à validação de requisitos é:
- a) priorizar os requisitos.
  - b) registrar as regras de negócio.
  - c) alocar requisitos aos subsistemas.





- d) definir critérios de aceitação do produto.
- e) especificar os atributos de qualidade do produto.

### Comentários:

(a) Errado. Priorizar os requisitos é uma prática associada à análise de requisitos, não à validação; (b) Errado. Registrar as regras de negócio é parte da especificação dos requisitos, não da validação; (c) Errado. Alocar requisitos aos subsistemas é parte da análise e design de sistemas, não da validação; (d) Correto. Definir critérios de aceitação do produto é uma prática fundamental na validação de requisitos, pois garante que os requisitos sejam claros e testáveis, permitindo verificar se o produto atende às expectativas dos stakeholders; (e) Errado. Especificar os atributos de qualidade do produto é parte da especificação dos requisitos, não da validação.

**Gabarito:** Letra D

**4. (CESGRANRIO / IPEA – 2024)** No projeto de desenvolvimento de um determinado software, uma equipe foi designada para criar, exclusivamente, testes funcionais e não funcionais para esse software. Nesse cenário, é recomendado que a equipe se envolva nesse projeto na etapa de:

- a) requisitos
- b) produção
- c) manutenção
- d) homologação
- e) implementação

### Comentários:

(a) Correto. Envolver a equipe de testes na etapa de requisitos é fundamental para garantir que todos os requisitos funcionais e não funcionais sejam bem compreendidos e possam ser testados adequadamente. Isso permite a criação de casos de teste que cobrem todos os aspectos do software desde o início do desenvolvimento;

(b) Errado. A etapa de produção é quando o software já está em uso pelos usuários finais. Envolver a equipe de testes apenas nesta fase seria muito tardio e ineficaz;

(c) Errado. A etapa de manutenção ocorre após o software ter sido lançado. Envolver a equipe de testes apenas nesta fase também seria muito tardio;

(d) Errado. A homologação é uma fase final de validação do software. Envolver a equipe de testes apenas nesta fase limita a capacidade de identificar e corrigir problemas precocemente;



(e) Errado. A implementação é a fase em que o código está sendo escrito. Embora a equipe de testes deva estar envolvida nesta fase, é fundamental que já tenham participado desde a etapa de requisitos para garantir uma cobertura de testes completa.

**Gabarito:** Letra A

5. **(CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento de um software para gerência de finanças pessoais decidiu adotar uma abordagem ágil, utilizando Histórias do Usuário para capturar requisitos funcionais. Essa técnica tem como característica descrever as funcionalidades do software do ponto de vista do usuário final. Para assegurar uma compreensão comum entre todos os envolvidos no projeto, é fundamental que a equipe entenda os componentes de uma História do Usuário. Os três componentes principais de uma História do Usuário são:

- a) Cartão, Conversação e Confirmação.
- b) Classes, Métodos e Atributos.
- c) Entidade, Relacionamento e Atributo.
- d) Requisitos Funcionais, Requisitos Não Funcionais e Requisitos de Domínio.
- e) Diagrama de Caso de Uso, Diagrama de Atividades e Diagrama de Sequência.

#### Comentários:

(a) Correto. Cartão, Conversação e Confirmação são componentes fundamentais de uma história do usuário;

(b) Errado. Esta alternativa está relacionada à programação orientada a objetos e não tem relação direta com as Histórias do Usuário na metodologia ágil;

(c) Errado. Esta alternativa refere-se a componentes de um modelo de dados entidade-relacionamento (ER) e não é aplicável às Histórias do Usuário;

(d) Errado. Estes são tipos de requisitos em um projeto de software, mas não são os componentes de uma História do Usuário na metodologia ágil;

(e) Errado. Estes são tipos de diagramas utilizados na modelagem UML (Unified Modeling Language) e não estão relacionados aos componentes de uma História do Usuário.

**Gabarito:** Letra A

6. **(CESGRANRIO / TRANSPETRO – 2023)** Os requisitos não funcionais podem ser divididos em três categorias: organizacionais, externos e de produto. Dentre os pertencentes à categoria organizacional estão os requisitos:

- a) legais



- b) operacionais
- c) reguladores
- d) de eficiência
- e) de usabilidade.

### Comentários:

(a) Errado. Requisitos legais pertencem à categoria de requisitos externos; (b) Correto. Requisitos operacionais são considerados organizacionais, pois dizem respeito ao ambiente operacional interno da organização; (c) Errado. Requisitos reguladores pertencem à categoria de requisitos externos; (d) Errado. Requisitos de eficiência pertencem à categoria de requisitos de produto; (e) Errado. Requisitos de usabilidade pertencem à categoria de requisitos de produto.

**Gabarito:** Letra B

**7. (CESGRANRIO / TRANSPETRO – 2023)** Ao desenvolver protótipos de alta fidelidade de um software, é necessário:

- a) aplicar a técnica de paper prototyping.
- b) realizar a primeira validação com algum usuário ou com algum cliente do protótipo criado.
- c) definir o wireframe das telas do software sem o uso das cores da solução.
- d) fornecer um layout básico que permita algumas validações iniciais de maneira digital.
- e) oferecer protótipos de telas que representam de maneira fiel o que é esperado pelos usuários para que assim sejam codificadas.

### Comentários:

(a) Errado. Paper prototyping é uma técnica para criar protótipos de baixa fidelidade utilizando papel e caneta, não para protótipos de alta fidelidade;

(b) Errado. Realizar a primeira validação pode ser parte do processo, mas a definição não é específica para protótipos de alta fidelidade;

(c) Errado. Definir wireframes sem cores se aplica a protótipos de baixa fidelidade, não de alta fidelidade;

(d) Errado. Fornecer um layout básico permite algumas validações iniciais, mas não caracteriza um protótipo de alta fidelidade;

(e) Correto. Protótipos de alta fidelidade oferecem telas que representam de maneira fiel o que é esperado pelos usuários, incluindo detalhes visuais e funcionais, para que assim possam ser codificadas.

**Gabarito:** Letra E



8. (VUNESP / TCM-SP – 2023) O estudo de viabilidade constitui-se na fase inicial do desenvolvimento da especificação de requisitos do software e o resultado desse estudo deve, prioritariamente, compreender:
- a) uma planilha de custos detalhada de todas as atividades e compras a serem feitas no decorrer do desenvolvimento do software.
  - b) a recomendação para prosseguir, ou não, com os demais processos componentes da engenharia de requisitos.
  - c) uma classificação detalhada dos requisitos levantados, por ordem de prioridade e de tempo necessários para o desenvolvimento.
  - d) a definição de quais deverão ser os módulos de software desenvolvidos internamente e quais deverão ser adquiridos de terceiros.
  - e) a definição da equipe de desenvolvimento do software, bem como sua seleção, segundo critérios estabelecidos nesse estudo.

#### Comentários:

- (a) Errado. A planilha de custos detalhada faz parte do planejamento do projeto, mas não é o foco principal do estudo de viabilidade;
- (b) Correto. O objetivo principal do estudo de viabilidade é determinar se vale a pena prosseguir com o desenvolvimento do software. Ele avalia se o projeto é viável técnica, econômica e operacionalmente, e emite uma recomendação para seguir adiante ou não com a engenharia de requisitos;
- (c) Errado. A classificação detalhada dos requisitos ocorre em etapas posteriores do processo de desenvolvimento de software, após a decisão de prosseguir com o projeto;
- (d) Errado. A definição de quais módulos serão desenvolvidos internamente ou adquiridos externamente é uma decisão de design ou planejamento, não o principal foco do estudo de viabilidade;
- (e) Errado. A definição da equipe de desenvolvimento e sua seleção é parte do planejamento do projeto, não do estudo de viabilidade.

**Gabarito:** Letra B

9. (VUNESP / UFABC – 2023) A Engenharia de Requisitos considera duas grandes vertentes: requisitos funcionais e requisitos não funcionais. Considerando os requisitos não funcionais, há



uma classificação que os divide nas categorias de produto, organizacionais e externos, sendo certo que requisitos:

- a) operacionais inserem-se na categoria de requisitos de produto.
- b) legais inserem-se na categoria de requisitos organizacionais.
- c) ambientais inserem-se na categoria de requisitos externos.
- d) de eficiência inserem-se na categoria de requisitos de produto.
- e) éticos inserem-se na categoria de requisitos de produto.

### Comentários:

(a) Errado, são requisitos organizacionais; (b) Errado, são requisitos externos; (c) Errado, são requisitos organizacionais; (d) Correto, são requisitos de produto; (e) Errado, são requisitos externos.

**Gabarito:** Letra D

**10. (CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento ágil decidiu utilizar Pontos de História (Story Points) para estimar o esforço necessário para implementar as funcionalidades descritas no seu Product Backlog, na forma de Histórias do Usuário. Considerando-se esse contexto, a técnica de Pontos de História ajuda a equipe a compreender o trabalho envolvido em cada história por:

- a) alocar tarefas específicas da história do usuário aos membros da equipe.
- b) classificar as histórias do usuário em categorias de prioridade, de alta a baixa.
- c) permitir prever o tempo, em horas, que a equipe levará para completar uma história.
- d) determinar o custo financeiro direto associado ao desenvolvimento de uma história do usuário.
- e) estimar o esforço relativo necessário para implementar uma determinada história do usuário, comparando-o ao de outras histórias.

### Comentários:

(a) Errado. Alocar tarefas específicas aos membros da equipe não é o objetivo dos Pontos de História;

(b) Errado. A classificação de histórias em categorias de prioridade é feita através de técnicas de priorização, não pela atribuição de Pontos de História;

(c) Errado. Pontos de História não se referem a tempo em horas, mas sim ao esforço relativo;



- (d) Errado. Pontos de História não determinam o custo financeiro direto do desenvolvimento;
- (e) Correto. Pontos de História são usados para estimar o esforço relativo necessário para implementar uma história do usuário, comparando-o ao esforço de outras histórias.

**Gabarito:** Letra E

**11. (CESGRANRIO / IPEA – 2024)** Um especialista em testes de software, com vasta experiência na criação de testes não funcionais, reuniu-se com sua equipe de desenvolvimento para avaliar se ainda havia requisitos não funcionais pendentes de teste em um novo software de e-commerce que estava sendo desenvolvido. Após ler a lista de requisitos identificados pela equipe como pendentes de teste, o especialista identificou um requisito não funcional ainda não testado. Com base no cenário apresentado, o requisito não funcional identificado foi o de verificar se:

- a) a opção esqueci senha, ao ser pressionada pelo usuário, o direcionava para uma tela que solicitava o e-mail do usuário para recuperar tal senha.
- b) o tempo de resposta para aprovar ou não alguma solicitação era de até 8 segundos.
- c) o chatbot do sistema está apresentando o conteúdo esperado, a partir de informações fornecidas por algum usuário em um chat de ajuda acessado pelo software.
- d) algum dos cupons de desconto de 30% para certos produtos escolhidos estava sendo aplicado corretamente.
- e) os relatórios de venda do mês estavam sendo exportados no formato PDF e CSV.

### Comentários:

- (a) Errado. Verificar a funcionalidade da opção "esqueci a senha" é um teste funcional, pois avalia se uma funcionalidade específica está funcionando corretamente;
- (b) Correto. Verificar o tempo de resposta é um teste de desempenho, que é um tipo de teste não funcional. Esse teste avalia se o sistema atende aos requisitos de desempenho especificados;
- (c) Errado. Verificar se o chatbot apresenta o conteúdo esperado é um teste funcional, pois está avaliando o comportamento esperado de uma funcionalidade específica;
- (d) Errado. Verificar a aplicação correta dos cupons de desconto é um teste funcional, pois avalia se uma funcionalidade específica está funcionando conforme esperado;
- (e) Errado. Verificar a exportação de relatórios em formatos específicos é um teste funcional, pois está testando uma funcionalidade específica do sistema.



**Gabarito:** Letra B

**12. (CESGRANRIO / IPEA – 2024)** A engenharia de requisitos pode ser dividida em gerenciamento de requisitos e desenvolvimento de requisitos, e este, por sua vez, pode ser subdividido em elicitação, análise, especificação e validação. Cada um desses quatro componentes do desenvolvimento de requisitos possui um conjunto de boas práticas. Uma das boas práticas associadas à validação de requisitos é:

- a) priorizar os requisitos.
- b) registrar as regras de negócio.
- c) alocar requisitos aos subsistemas.
- d) definir critérios de aceitação do produto.
- e) especificar os atributos de qualidade do produto.

**Comentários:**

(a) Errado. Priorizar os requisitos é uma prática associada à análise de requisitos, não à validação; (b) Errado. Registrar as regras de negócio é parte da especificação dos requisitos, não da validação; (c) Errado. Alocar requisitos aos subsistemas é parte da análise e design de sistemas, não da validação; (d) Correto. Definir critérios de aceitação do produto é uma prática fundamental na validação de requisitos, pois garante que os requisitos sejam claros e testáveis, permitindo verificar se o produto atende às expectativas dos stakeholders; (e) Errado. Especificar os atributos de qualidade do produto é parte da especificação dos requisitos, não da validação.

**Gabarito:** Letra D

**13. (CESGRANRIO / IPEA – 2024)** No projeto de desenvolvimento de um determinado software, uma equipe foi designada para criar, exclusivamente, testes funcionais e não funcionais para esse software. Nesse cenário, é recomendado que a equipe se envolva nesse projeto na etapa de:

- a) requisitos
- b) produção
- c) manutenção
- d) homologação
- e) implementação

**Comentários:**

(a) Correto. Envolver a equipe de testes na etapa de requisitos é fundamental para garantir que todos os requisitos funcionais e não funcionais sejam bem compreendidos e possam ser testados adequadamente. Isso permite a criação de casos de teste que cobrem todos os aspectos do software desde o início do desenvolvimento;





- (b) Errado. A etapa de produção é quando o software já está em uso pelos usuários finais. Envolver a equipe de testes apenas nesta fase seria muito tardio e ineficaz;
- (c) Errado. A etapa de manutenção ocorre após o software ter sido lançado. Envolver a equipe de testes apenas nesta fase também seria muito tardio;
- (d) Errado. A homologação é uma fase final de validação do software. Envolver a equipe de testes apenas nesta fase limita a capacidade de identificar e corrigir problemas precocemente;
- (e) Errado. A implementação é a fase em que o código está sendo escrito. Embora a equipe de testes deva estar envolvida nesta fase, é fundamental que já tenham participado desde a etapa de requisitos para garantir uma cobertura de testes completa.

**Gabarito:** Letra A

- 14. (CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento de um software para gerência de finanças pessoais decidiu adotar uma abordagem ágil, utilizando Histórias do Usuário para capturar requisitos funcionais. Essa técnica tem como característica descrever as funcionalidades do software do ponto de vista do usuário final. Para assegurar uma compreensão comum entre todos os envolvidos no projeto, é fundamental que a equipe entenda os componentes de uma História do Usuário. Os três componentes principais de uma
- História do Usuário são:
    - a) Cartão, Conversação e Confirmação.
    - b) Classes, Métodos e Atributos.
    - c) Entidade, Relacionamento e Atributo.
    - d) Requisitos Funcionais, Requisitos Não Funcionais e Requisitos de Domínio.
    - e) Diagrama de Caso de Uso, Diagrama de Atividades e Diagrama de Sequência.

### Comentários:

- (a) Correto. Cartão, Conversação e Confirmação são componentes fundamentais de uma história do usuário;
- (b) Errado. Esta alternativa está relacionada à programação orientada a objetos e não tem relação direta com as Histórias do Usuário na metodologia ágil;
- (c) Errado. Esta alternativa refere-se a componentes de um modelo de dados entidade-relacionamento (ER) e não é aplicável às Histórias do Usuário;
- (d) Errado. Estes são tipos de requisitos em um projeto de software, mas não são os componentes de uma História do Usuário na metodologia ágil;



(e) Errado. Estes são tipos de diagramas utilizados na modelagem UML (Unified Modeling Language) e não estão relacionados aos componentes de uma História do Usuário.

**Gabarito:** Letra A

**15. (CESGRANRIO / TRANSPETRO – 2023)** Os requisitos não funcionais podem ser divididos em três categorias: organizacionais, externos e de produto. Dentre os pertencentes à categoria organizacional estão os requisitos:

- a) legais
- b) operacionais
- c) reguladores
- d) de eficiência
- e) de usabilidade.

**Comentários:**

(a) Errado. Requisitos legais pertencem à categoria de requisitos externos; (b) Correto. Requisitos operacionais são considerados organizacionais, pois dizem respeito ao ambiente operacional interno da organização; (c) Errado. Requisitos reguladores pertencem à categoria de requisitos externos; (d) Errado. Requisitos de eficiência pertencem à categoria de requisitos de produto; (e) Errado. Requisitos de usabilidade pertencem à categoria de requisitos de produto.

**Gabarito:** Letra B

**16. (CESGRANRIO / TRANSPETRO – 2023)** Ao desenvolver protótipos de alta fidelidade de um software, é necessário:

- a) aplicar a técnica de paper prototyping.
- b) realizar a primeira validação com algum usuário ou com algum cliente do protótipo criado.
- c) definir o wireframe das telas do software sem o uso das cores da solução.
- d) fornecer um layout básico que permita algumas validações iniciais de maneira digital.
- e) oferecer protótipos de telas que representam de maneira fiel o que é esperado pelos usuários para que assim sejam codificadas.

**Comentários:**

(a) Errado. Paper prototyping é uma técnica para criar protótipos de baixa fidelidade utilizando papel e caneta, não para protótipos de alta fidelidade;

(b) Errado. Realizar a primeira validação pode ser parte do processo, mas a definição não é específica para protótipos de alta fidelidade;



- (c) Errado. Definir wireframes sem cores se aplica a protótipos de baixa fidelidade, não de alta fidelidade;
- (d) Errado. Fornecer um layout básico permite algumas validações iniciais, mas não caracteriza um protótipo de alta fidelidade;
- (e) Correto. Protótipos de alta fidelidade oferecem telas que representam de maneira fiel o que é esperado pelos usuários, incluindo detalhes visuais e funcionais, para que assim possam ser codificadas.

**Gabarito:** Letra E

**17. (VUNESP / TCM-SP – 2023)** O estudo de viabilidade constitui-se na fase inicial do desenvolvimento da especificação de requisitos do software e o resultado desse estudo deve, prioritariamente, compreender:

- a) uma planilha de custos detalhada de todas as atividades e compras a serem feitas no decorrer do desenvolvimento do software.
- b) a recomendação para prosseguir, ou não, com os demais processos componentes da engenharia de requisitos.
- c) uma classificação detalhada dos requisitos levantados, por ordem de prioridade e de tempo necessários para o desenvolvimento.
- d) a definição de quais deverão ser os módulos de software desenvolvidos internamente e quais deverão ser adquiridos de terceiros.
- e) a definição da equipe de desenvolvimento do software, bem como sua seleção, segundo critérios estabelecidos nesse estudo.

#### **Comentários:**

- (a) Errado. A planilha de custos detalhada faz parte do planejamento do projeto, mas não é o foco principal do estudo de viabilidade;
- (b) Correto. O objetivo principal do estudo de viabilidade é determinar se vale a pena prosseguir com o desenvolvimento do software. Ele avalia se o projeto é viável técnica, econômica e operacionalmente, e emite uma recomendação para seguir adiante ou não com a engenharia de requisitos;
- (c) Errado. A classificação detalhada dos requisitos ocorre em etapas posteriores do processo de desenvolvimento de software, após a decisão de prosseguir com o projeto;





(V) Correto. Essas entrevistas são úteis para obter um entendimento geral sobre o que os stakeholders fazem; (V) Correto. Brainstorming (também conhecida como Tempestade de Ideias) ocorre em um ambiente informal, buscando explorar a potencialidade criativa de um grupo; (V) Correto. É uma técnica que é utilizada no estágio inicial do projeto e ajuda os stakeholders a desenvolverem uma noção sobre a aplicação que irá ser implementada.

**Gabarito:** Letra E

**20. (IDIB / CRF – MS – 2021)** Em qual etapa do processo de desenvolvimento de requisitos de software mais comumente costuma acontecer a identificação de gaps nos requisitos ou a identificação de requisitos desnecessários, conforme eles se relacionam com o escopo definido?

- a) Elicitação de requisitos.
- b) Análise de requisitos.
- c) Especificação de requisitos.
- d) Validação de requisitos.

#### Comentários:

Bem, o nome das fases vai variar de autor para autor (e, às vezes, eles mesmos mudam de ideia entre um livro e outro), mas dá para responder por bom senso. A etapa do processo de desenvolvimento de requisitos de software mais comumente costuma acontecer a identificação de gaps nos requisitos ou a identificação de requisitos desnecessários é a Análise de Requisitos. Ela ocorre logo após a Elicitação de Requisitos, que é responsável por identificar os requisitos.

**Gabarito:** Letra B

**21. (IDIB / CRF - MS – 2021)** "Uma descrição de uma propriedade ou característica que um sistema deve exibir ou uma restrição que ele deve respeitar". Tal definição se adequa a qual tipo de requisito de informação em um processo de software?

- a) Requisito Funcional.
- b) Requisito Não-Funcional.
- c) Atributo de Qualidade.
- d) Requisito do Usuário.

#### Comentários:

Os requisitos não-funcionais são restrições ou condições estipuladas sobre as quais o sistema deve funcionar. Além disso, eles podem descrever características do sistema como confiabilidade, segurança, usabilidade, performance, etc.



**Gabarito:** Letra B

**22. (VUNESP / EBSE RH – 2020)** Considerando as técnicas utilizadas para a avaliação de requisitos, é correto afirmar que, na:

- a) facilidade de verificação, deve-se verificar se não há requisitos conflitantes entre si.
- b) verificação de consistência, deve-se verificar se os requisitos podem ser implementados, considerando a tecnologia existente.
- c) verificação de realismo, deve-se verificar se todas as funções e restrições planejadas estão contempladas.
- d) verificação de validade, deve-se verificar se não há requisitos conflitantes entre si.
- e) verificação de completeza, deve-se verificar se todas as funções e restrições planejadas estão contempladas.

**Comentários:**

(a) Errado, essa é a definição da verificação de consistência; (b) Errado, essa é a definição de verificação de realismo; (c) Errado, essa é a definição de verificação de completeza; (d) Errado, essa é a definição de verificação de consistência; (e) Correto, é a definição correta de verificação de completeza.

**Gabarito:** Letra E

**23. (VUNESP / EBSE RH – 2020)** Na engenharia de requisitos, um fator importante são os requisitos não funcionais, que se classificam em organizacionais, de produto e externos. Os requisitos

- a) de produto têm origem em políticas e procedimentos da organização do cliente.
- b) de produto compreendem fatores oriundos de fatores externos ao sistema e seu processo de desenvolvimento.
- c) externos especificam o comportamento do produto, tais como o desempenho e a memória requerida.
- d) organizacionais têm origem em políticas e procedimentos da organização do cliente.
- e) organizacionais especificam o comportamento do produto, tais como o desempenho e a memória requerida.

**Comentários:**

(a) Errado, trata-se da definição de requisitos organizacionais; (b) Errado, trata-se da definição de requisitos externos; (c) Errado, trata-se da definição de requisitos de produto; (d) Correto, os requisitos organizacionais são derivados de políticas e procedimentos da organização do cliente e do desenvolvedor; (e) Errado, trata-se da definição de requisitos de produto.

**Gabarito:** Letra D



**24. (COPESE - UFPI / ALEPI – 2020)** Um técnico de TI da ALEPI que gerencia uma equipe de desenvolvimento de software na Assembleia, eventualmente necessita fazer levantamento de requisitos da aplicação que está sendo desenvolvida. Sobre os Requisitos de Software, considere as seguintes afirmativas:

I. Requisitos funcionais são aqueles que definem parte da funcionalidade do sistema e podem ser categorizados em três tipos: requisitos de produtos, requisitos organizacionais e requisitos externos.

II. Requisitos não-funcionais dizem respeito a restrições, aspectos de desempenho, interfaces com o usuário, confiabilidade, segurança, portabilidade e padrões.

III. Requisitos organizacionais estão relacionados às metas da empresa, suas políticas estratégicas adotadas, assim como assuntos relacionados aos empregados da empresa com seus respectivos objetivos.

IV. Requisitos de produto estão relacionados as restrições impostas por fatores externos ao sistema tais como restrições de interoperabilidade, éticas e legais.

Marque a opção que corresponde somente às afirmativas verdadeiras.

- a) Apenas I, III e IV.
- b) Apenas I, II e III.
- c) Apenas I e III.
- d) Apenas II e III.
- e) Apenas II, III e IV.

#### Comentários:

(I) Errado. Apesar de a primeira parte da afirmação estar correta, a parte final refere-se a Requisitos Não-Funcionais; (II) Correto. Requisitos Não-Funcionais são restrições ou condições estipuladas sobre as quais o sistema deve funcionar; (III) Correto. Esse item define corretamente os Requisitos Organizacionais; (IV) Errado. Na verdade, a definição é de Requisitos Externos.

**Gabarito:** Letra D

**25. (FAFIPA / Prefeitura de Arapongas - PR – 2020)** A Engenharia de Requisitos é um termo cunhado para descrever as atividades relacionadas à investigação e definição de escopo de um sistema de software, ou seja, trata-se do processo de descobrir, analisar, documentar e verificar as funções e restrições do sistema. Para auxiliar o levantamento de requisitos, existe um conjunto de técnicas de levantamento de dados que podem ser aplicadas em conjunto ou





isoladamente, a depender das características do projeto. Assinale a alternativa que apresenta somente técnicas para descoberta de requisitos:

- a) Sprint; Caso de Uso; Etnografia.
- b) Entrevista; Caso de Uso; Etnografia.
- c) Sprint; Refatoração; Etnografia.
- d) JAD; Refatoração; Etnografia.
- e) Entrevista; JAD; Refatoração.

### Comentários:

De todos os itens citados, apenas Sprint e Refatoração não são técnicas para descoberta de requisitos.

**Gabarito:** Letra B

**26. (IBFC / EBSEH – 2020)** Requisitos são as bases para todo projeto, definindo o que as partes interessadas de um novo sistema necessitam e também o que o sistema deve fazer para satisfazer as suas necessidades. Antes do processo, propriamente dito, da Engenharia de Requisitos deve-se ter:

- a) entrevistas e questionários com os usuários
- b) a documentação dos requisitos
- c) a revisão dos requisitos funcionais e não-funcionais
- d) os estudos de viabilidade técnica/financeira
- e) a revisão dos requisitos pelos usuários

### Comentários:

(a) Errado, as entrevistas ocorrem na fase de Elicitação; (b) Errado, a documentação ocorre na fase de Elicitação; (c) Errado, a revisão dos requisitos ocorre na fase de Validação dos Requisitos; (d) Correto, os estudos de viabilidade ocorrem na primeira fase da Engenharia de Requisitos, nela ocorre uma avaliação rápida e de baixo custo para verificar se as necessidades dos usuários podem ser satisfeitas; (e) Errada, a revisão de requisitos ocorre na fase de Validação dos Requisitos.

**Gabarito:** Letra D

**27. (INSTITUTO AOCP / Prefeitura de Betim - MG – 2020)** A engenharia de requisitos estabelece uma ponte entre o projeto e a construção do software. Assinale a alternativa que representa a etapa na qual as metas de negócio são estabelecidas.

- a) Levantamento.
- b) Concepção.



- c) Negociação.
- d) Revisão.
- e) Gestão.

**Comentários:**

Pessoal, é na etapa de Levantamento de Requisitos que se busca saber os requisitos que o sistema deve ter, ou seja, o que o cliente deseja que tenha em seu sistema. Além disso, os stakeholders que participam da etapa perguntam ao cliente qual é o objetivo do produto, como o produto se enquadra nas necessidades do negócio e como o produto será utilizado. A partir disso são estabelecidas as metas do negócio.

**Gabarito:** Letra A

**28.(CESGRANRIO / TRANSPETRO – 2018)** Os requisitos não funcionais são requisitos que não estão diretamente relacionados com o que o sistema é capaz de fazer para atender aos seus usuários. Eles podem estar relacionados a algumas características, propriedades ou restrições como confiabilidade, ambiente de operação, aspectos legais e outros. Alguns autores os classificam em requisitos de produto, organizacionais e externos.

Os requisitos de produto ainda são classificados como de usabilidade, confiança, proteção e:

- a) ambientais
- b) operacionais
- c) segurança
- d) éticos
- e) eficiência

**Comentários:**



Conforme podemos ver pela figura, os requisitos de produto são divididos em: requisitos de confiança, requisitos de proteção, requisitos de usabilidade e... requisitos de eficiência.



**Gabarito:** Letra E

**29.(CESGRANRIO / TRANSPETRO – 2018)** Ao realizar uma especificação de requisitos, um analista resolveu utilizar a técnicas QFD. Nessa técnica, os requisitos esperados têm como característica ser:

- a) explícitos e causar insatisfação significativa se não implementados.
- b) implícitos e causar insatisfação significativa se não implementados.
- c) explícitos e causar satisfação se implementados.
- d) implícitos e causar satisfação se implementados.
- e) desconhecidos e causar satisfação significativa se implementados.

**Comentários:**

(a) Errado. Refere-se a requisitos claramente expressos pelos clientes, cuja ausência pode resultar em uma avaliação negativa do produto;

(b) Correto. Refere-se a requisitos que não são diretamente expressos pelos clientes, mas que, se não atendidos, podem levar a insatisfação;

(c) Errado. Refere-se a requisitos claramente declarados pelos usuários, cuja implementação pode aumentar a satisfação, mas sua ausência não necessariamente leva à insatisfação;

(d) Errado. Corresponde aos requisitos não expressamente solicitados pelos clientes, mas que, se presentes, podem surpreender positivamente e aumentar a satisfação com o produto;

(e) Errado. Refere-se a requisitos que os clientes não têm consciência de precisar, mas que, se implementados, podem gerar um alto nível de satisfação.

**Gabarito:** Letra B

**30.(CESGRANRIO / TRANSPETRO – 2018)** A técnica de etnografia para o levantamento de requisitos implica a:

- a) imersão do analista no ambiente de trabalho onde o sistema vai ser usado.
- b) imersão do usuário no ambiente de desenvolvimento.
- c) realização de reuniões de levantamento de requisitos com grupos de usuários de um mesmo tipo.
- d) realização de reuniões de levantamento de requisitos com grupos de usuários de tipos diferentes.
- e) realização de reuniões de levantamento no ambiente de trabalho do usuário.



### Comentários:

- (a) Correto Essa é a essência da etnografia, que é a observação e participação do pesquisador no ambiente natural dos usuários para compreender suas práticas, desafios e necessidades;
- (b) Errado. abordagem pode ser útil para feedback sobre protótipos ou para entender melhor as solicitações dos usuários, mas não é característica da etnografia;
- (c) Errado. Embora útil para coletar informações, isso não se alinha diretamente com a metodologia etnográfica, que enfatiza a observação no ambiente natural em vez de reuniões estruturadas;
- (d) Errado. Tal como apresentado na alternativa anterior, isso não reflete uma abordagem etnográfica;
- (e) Errado. Embora essa alternativa se aproxime do ambiente natural do usuário, ela não captura totalmente a imersão e a observação contínua que define a etnografia.

**Gabarito:** Letra A

**31. (CESGRANRIO / TRANSPETRO – 2018)** Há autores que classificam os requisitos não funcionais de um sistema de informação em requisitos de produto, requisitos organizacionais e requisitos externos. Cada uma dessas categorias também sofre subdivisões. Considerando essa informação e o desenvolvimento de um sistema acadêmico, são pares que estabelecem a correspondência (categoria de requisito; exemplo), EXCETO:

- a) (eficiência; os relatórios de notas devem ser impressos em menos de cinco segundos).
- b) (segurança; o sistema deve prover recursos para registro de ponto do professor por reconhecimento de impressão digital).
- c) (ética; cada trabalho proposto possuirá um único prazo para todos os alunos do curso).
- d) (usabilidade; o professor lança as faltas dos alunos das turmas nas quais leciona).
- e) (ambiental; os servidores devem trabalhar entre 40% e 55% de umidade relativa do ar e 5° a 15° de temperatura).

### Comentários:

- (a) Errado. Este é um exemplo de requisito de produto relacionado à eficiência, pois especifica a performance do sistema em termos de tempo de resposta para uma tarefa específica;



- b) Errado. Este também é um requisito de produto, especificamente de segurança, garantindo que o acesso ao sistema seja controlado e autenticado de forma segura;
- c) Errado. Esta alternativa parece deslocada. Enquanto a ética é uma consideração importante no desenvolvimento de sistemas, ela não é comumente listada como uma categoria de requisito não funcional na literatura tradicional de engenharia de software. Requisitos éticos, como tratados aqui, seriam mais apropriadamente discutidos em termos de políticas organizacionais ou práticas de gestão educacional, mas não como requisitos não funcionais de um sistema;
- d) Correto. Isso se enquadra na categoria de usabilidade, um requisito de produto que especifica a facilidade com que os usuários podem usar o sistema para completar suas tarefas;
- e) Errado. Isso é um exemplo de um requisito externo, especificamente um requisito ambiental, que define as condições físicas sob as quais o sistema deve operar eficientemente.

**Gabarito:** Letra D

**32. (CESGRANRIO / Banco da Amazônia – 2018)** Requisitos existem em vários níveis de abstração. Um desses níveis é conhecido como “requisitos de negócio”, os quais:

- a) definem a visão e o escopo que influenciam os requisitos de usuário.  
b) são dependentes dos requisitos funcionais para serem levantados.  
c) são descobertos depois e a partir dos requisitos de usuário.  
d) são registrados na forma da Especificação de Requisitos de Software.  
e) influenciam e são registrados na forma de regras de negócio.

### Comentários:

- (a) Correto. Os requisitos de negócio definem a visão e o escopo do projeto, influenciando diretamente os requisitos de usuário. Eles estabelecem o que a organização espera alcançar com o projeto, orientando as necessidades e os objetivos dos usuários finais;
- (b) Errado. Os requisitos de negócio não são dependentes dos requisitos funcionais; na verdade, é o contrário. Os requisitos funcionais, que detalham o comportamento específico do sistema, são derivados dos requisitos de negócio;
- (c) Errado. Os requisitos de negócio são tipicamente identificados antes dos requisitos de usuário. Eles fornecem a base para o desenvolvimento dos requisitos de usuário, que detalham as necessidades específicas dos usuários finais com base nos objetivos de negócio;
- (d) Errado. Embora os requisitos de negócio possam influenciar a Especificação de Requisitos de Software (SRS), eles não são registrados como tal. A SRS normalmente contém requisitos de



sistema e requisitos de usuário, detalhando como o software deve funcionar para atender a esses requisitos;

(e) Errado. Embora os requisitos de negócio possam influenciar as regras de negócio, eles não são, por si só, registrados apenas como regras de negócio. As regras de negócio são diretrizes específicas que governam as operações, decisões e procedimentos da organização, que podem ser derivadas dos requisitos de negócio.

**Gabarito:** Letra A

**33. (CESGRANRIO / Petrobras – 2018)** O processo de desenvolvimento de requisitos pode ser dividido em 4 etapas: elicitação, análise, especificação e validação. Qual prática da Engenharia de Requisitos está associada à etapa de análise?

- a) Criar protótipos.
- b) Identificar casos de uso.
- c) Inspeccionar documentos de requisitos.
- d) Definir critérios de aceitação.
- e) Registrar regras de negócio.

#### Comentários:

(a) Correto. A criação de protótipos é uma técnica usada principalmente na elicitação e validação de requisitos para ajudar a esclarecer, demonstrar e validar requisitos com as partes interessadas. Ian Sommerville afirma que protótipos podem fazer parte da etapa de Elicitação e Análise de Requisitos, daí eu acredito que a questão considerou os protótipos como parte da análise;

(b) Errado. A identificação de casos de uso é uma prática associada à elicitação de requisitos; na análise de requisitos, eles buscam detalhar como os usuários interagem com o sistema e quais são as funcionalidades necessárias;

(c) Errado. A inspeção de documentos é uma atividade importante, mas está mais alinhada com a validação de requisitos, onde os requisitos documentados são revisados para assegurar sua qualidade e completude;

(d) Errado. Definir critérios de aceitação é uma prática importante, mas ela está mais diretamente relacionada à validação de requisitos e ao planejamento de testes, servindo como base para verificar se o sistema atende aos requisitos especificados;

(e) Errado. Registrar regras de negócio é uma atividade que pode ocorrer tanto na especificação quanto na análise de requisitos. No contexto da análise, as regras de negócio ajudam a entender as políticas, condições e restrições sob as quais o sistema deve operar, contribuindo para a análise e modelagem dos requisitos.



**34. (CESGRANRIO / Petrobras – 2018)** Um analista em treinamento apresenta a seguinte descrição para alguns requisitos de um sistema on-line para venda de bilhetes de uma rede de cinemas em que cada cinema apresenta mais de um filme, e cada filme é apresentado em mais de um cinema com lugares marcados. Quando o usuário entrar no site da rede, um menu é apresentado com os filmes e cinemas disponíveis. O usuário seleciona o filme a que deseja assistir e entra com o número de cartão de crédito. A validade do cartão é verificada e o bilhete é emitido. O sistema encerra a operação com a mensagem: obrigado, divirta-se. Embora haja falhas nessa descrição ainda incipiente, o instrutor solicitou que o analista a estendesse com exemplos de possíveis requisitos não funcionais para o sistema.

Qual é a sentença que NÃO atende à solicitação do instrutor?

- a) O tempo máximo de espera do sistema para a inserção do número do cartão de crédito deve ser maior do que oito segundos e menor do que 15 segundos.
- b) O sistema deve estar disponível por 99,5% do tempo durante as 24 horas do dia, sete dias por semana.
- c) O formato do bilhete deve seguir o padrão definido na legislação municipal pertinente.
- d) O usuário deve selecionar o cinema em que deseja assistir ao filme.
- e) O sistema deve ser compatível com operação em smartphone.

#### Comentários:

- (a) Errado. Este é um exemplo de requisito não funcional relacionado ao desempenho e tempo de resposta do sistema, especificando quanto tempo o sistema deve esperar pela ação do usuário;
- (b) Errado. Este requisito não funcional se refere à disponibilidade do sistema, estabelecendo uma meta quantitativa para o tempo que o sistema deve estar operacional e acessível aos usuários;
- (c) Errado. Este é um requisito não funcional de conformidade, indicando que o sistema deve aderir a normas legais externas quanto ao formato dos bilhetes emitidos;
- (d) Correto. Esta sentença descreve um requisito funcional, não um não funcional, pois especifica uma ação que o sistema deve permitir que o usuário faça (um comportamento do sistema);
- (e) Errado. Este requisito não funcional se refere à usabilidade e compatibilidade, estabelecendo que o sistema deve ser acessível e utilizável em dispositivos móveis.

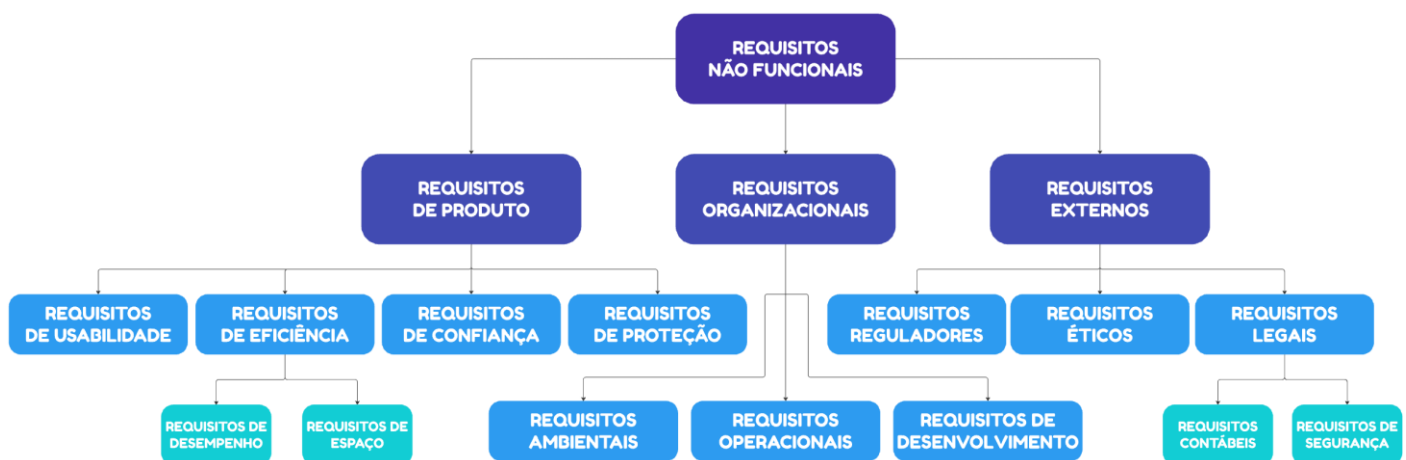




35. (FAURGS / TJ-RS – 2018) Requisitos não funcionais – como o nome sugere – são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Podem ser provenientes das características requeridas para o software, da organização que desenvolve o software ou de fontes externas. Os requisitos não funcionais que especificam ou restringem o comportamento do software – como por exemplo o seu desempenho, seus requisitos de proteção, seus requisitos de usabilidade e a taxa aceitável de falhas – são denominados requisitos:

- a) organizacionais.
- b) de produto.
- c) externos.
- d) éticos.
- e) ambientais.

Comentários:



São os Requisitos de Produto: Confiabilidade, Proteção, Eficiência (Desempenho e Armazenamento) e Usabilidade.

36. (FAURGS / TJ-RS – 2018) Técnicas de descoberta de requisitos (às vezes chamada de elicitación de requisitos) é o processo de reunir informações sobre o sistema requerido e os sistemas existentes e separar dessas informações os requisitos do usuário e de sistema; o uso destas técnicas faz parte da maioria dos processos de engenharia de requisitos. \_\_\_\_\_ é o nome dado a uma técnica em que o stakeholder responde a um conjunto predefinido de perguntas sobre o sistema usado no momento e sobre o sistema que será desenvolvido; os requisitos surgem a partir das respostas a essas perguntas.



A alternativa que contém o termo que completa corretamente a lacuna do texto acima é:

- a) Entrevista fechada.
- b) Entrevista aberta.
- c) Etnografia.
- d) Cenários.
- e) Casos de uso.

#### Comentários:

A palavra-chave pra essa questão é "**predefinido**". Se é um conjunto de perguntas a serem respondidas por um stakeholder, é uma entrevista. Se essa entrevista já possui um conjunto predefinido de perguntas, ela é uma entrevista fechada.

**Gabarito:** Letra A

**37. (FAURGS / TJ-RS – 2018)** Qual alternativa abaixo apresenta um requisito funcional de software?

- a) A base de dados deve ser protegida para acesso apenas a usuários autorizados.
- b) O tempo de resposta do sistema não deve ultrapassar 30 segundos..
- c) O software deve ser operacionalizado no Sistema Operacional Windows.
- d) O software deve emitir relatórios de vendas.
- e) O tempo de desenvolvimento não deve ultrapassar três meses.

#### Comentários:

(a) Errado, isso é uma restrição a uma funcionalidade, logo é um RNF; (b) Errado, isso é uma restrição a uma funcionalidade, logo é um RNF; (c) Errado, isso é uma restrição a uma funcionalidade, logo é um RNF; (d) Correto, isso é uma funcionalidade a ser oferecida pelo sistema, logo é um RF; (e) Errado, isso é uma restrição a uma funcionalidade, logo é um RNF;

**Gabarito:** Letra D

**38. (FAURGS / TJ-RS – 2018)** Considerando que, durante o processo de validação de requisitos, estes são submetidos a diferentes tipos de verificação, assinale a alternativa cuja verificação indica que no documento de requisitos não existem descrições diferentes para uma mesma função do sistema.

- a) Verificação de consistência.
- b) Verificação de completude.
- c) Verificações de realismo.
- d) Verificações de validade.
- e) Prototipação.



### Comentários:

Galera, essa dá para responder por bom senso! Se a questão diz que é uma verificação de que não existem descrições **diferentes** para uma **mesma** função, só pode ser uma verificação de consistência.

**Gabarito:** Letra A

**39.(UFG / SANEAGO – 2017)** Com relação à qualidade de software, é um exemplo de requisito de software não funcional:

- a) calcular o valor do desconto conforme o perfil do cliente.
- b) processar até 100 pedidos por segundo.
- c) produzir o software em até dois anos e ter custo inferior a R\$100.000,00.
- d) limitar o cadastro de usuários ao Departamento de Pessoal.

### Comentários:

(a) Errado, isso é uma funcionalidade, logo se trata de um RF; (b) Correto, isso é uma restrição a uma funcionalidade, logo é um RNF; (c) Errado, isso é um requisito de projeto e, não, de software; (d) Errado, isso é uma configuração e, não, um requisito de software.

**Gabarito:** Letra B

**40.(UFG / SANEAGO – 2017)** Alguns usuários estão insatisfeitos com um software. Uma investigação revelou que a origem da insatisfação decorre de uma omissão (requisito não especificado) na especificação de requisitos de software. Que atividade da engenharia de requisitos precisa ser revista para evitar problemas semelhantes?

- a) Análise de viabilidade.
- b) Validação.
- c) Construção.
- d) Elaboração do Termo de Abertura.

### Comentários:

A atividade a ser revista é a Validação! Essa é a atividade responsável por encontrar erros de interpretação, ambiguidades e omissões.

**Gabarito:** Letra B

**41.(UFG / SANEAGO – 2017)** São atributos de requisito de software:



- a) custo (para implementar) e complexidade ciclométrica.
- b) prioridade e linguagem de implementação.
- c) complexidade ciclométrica e risco.
- d) risco e identificador.

### Comentários:

Honestamente, não sei de onde essa questão foi retirada, mas é possível responder por eliminação: (a) Errado, Complexidade Ciclométrica é uma métrica; (b) Errado, requisitos são independentes de linguagem de programação; (c) Complexidade Ciclométrica é uma métrica; (d) Correto, risco e identificador.

**Gabarito:** Letra D

**42. (UFG / SANEAGO – 2017)** Uma Engenharia de Requisitos (ER) bem estruturada garante qualidade, confiabilidade e integridade ao produto de software a ser desenvolvido. O conjunto de atributos que evidencia o esforço necessário para fazer modificações especificadas no software é uma característica de:

- a) portabilidade.
- b) confiabilidade.
- c) manutenibilidade.
- d) eficiência.

### Comentários:

Em engenharia de software, manutenibilidade é um aspecto da qualidade de software que se refere à facilidade de um software de ser modificado a fim de corrigir defeitos, adequar-se a novos requisitos, aumentar a suportabilidade ou se adequar a um ambiente novo. Era possível responder essa por bom senso! :)

**Gabarito:** Letra C

**43. (IBFC / EBSEH – 2017)** Quanto aos vários tipos de requisitos assinale, das alternativas abaixo, a única que NÃO identifica corretamente um clássico requisito não-funcional:

- a) requisito de implementação da arquitetura do sistema
- b) requisitos de funcionalidades do sistema
- c) requisito de interoperabilidade da arquitetura do sistema
- d) requisitos de confiabilidade da arquitetura do sistema
- e) requisitos de portabilidade da arquitetura do sistema



### Comentários:

(a) Errado, requisitos de arquitetura realmente são não-funcionais; (b) Correto, requisitos de funcionalidades do sistema definitivamente não são requisitos não-funcionais; (c) Errado, requisitos de interoperabilidade realmente são não-funcionais; (d) Errado, requisitos de confiabilidade da arquitetura realmente são não-funcionais; (e) Errado, requisitos de portabilidade da arquitetura realmente são não-funcionais.

**Gabarito:** Letra B

**44. (IBFC / EBSERH – 2017)** A Análise de Requisitos é a primeira fase de desenvolvimento de software dividido em Requisitos funcionais e Requisitos não-funcionais. Os Requisitos não-funcionais possuem vários tipos diferentes de classificação tais como:

- (1) Requisitos de confiabilidade.
- (2) Requisitos de produtos.
- (3) Requisitos éticos.
- (4) Requisitos de portabilidade.

- a) da relação apresentada existem somente o 2, 3 e 4
- b) da relação apresentada existem somente o 1, 3 e 4
- c) da relação apresentada existem somente o 1, 2 e 4
- d) da relação apresentada existem somente o 1, 2 e 3
- e) da relação apresentada existem todos.

### Comentários:

(1) Requisitos de Confiabilidade são RNF; (2) Requisitos de Produtos são RNF; (3) Requisitos Éticos são RNF; (4) Requisitos de Portabilidade são RNF. Logo, todos eles são RNF.

**Gabarito:** Letra E

**45. (CESGRANRIO / IBGE – 2016)** Um dos objetivos da disciplina de requisitos é:

- a) criar um esboço inicial da arquitetura do sistema a ser desenvolvido.
- b) adaptar e configurar o processo de desenvolvimento de modo a atender às especificidades do sistema a ser desenvolvido.
- c) fornecer uma base para estimar o custo e o tempo de desenvolvimento de um sistema.
- d) assegurar que os clientes, os usuários e os desenvolvedores tenham um entendimento comum da organização na qual um sistema será implantado.



e) entender a estrutura e a dinâmica da organização na qual um sistema será implantado.

### Comentários:

(a) Errado. Embora a compreensão dos requisitos possa influenciar a arquitetura do sistema, o foco principal da disciplina de requisitos não é criar um esboço da arquitetura, mas sim definir o que o sistema deve fazer;

(b) Errado. Isso é mais relacionado ao gerenciamento de projetos e à engenharia de software em geral do que à disciplina de requisitos especificamente;

(c) Correto. Esta é uma função crítica da disciplina de requisitos. A compreensão clara e detalhada dos requisitos é essencial para estimar com precisão o custo e o tempo necessários para desenvolver o sistema. Requisitos bem definidos permitem uma melhor previsão dos recursos necessários e dos prazos de entrega;

(d) Errado. Embora o entendimento comum da organização seja importante, o foco principal da disciplina de requisitos é garantir um entendimento comum do sistema a ser desenvolvido, mais do que da organização em si;

(e) Errado. Isso pode ser parte do contexto de análise de requisitos, mas não é o objetivo principal da disciplina. O foco está mais em entender as necessidades e os requisitos do sistema do que a estrutura organizacional per se.

**Gabarito:** Letra C

**46.(CESGRANRIO / IBGE – 2016)** Um dos objetivos da disciplina de requisitos é:

a) criar um esboço inicial da arquitetura do sistema a ser desenvolvido.

b) adaptar e configurar o processo de desenvolvimento de modo a atender às especificidades do sistema a ser desenvolvido.

c) fornecer uma base para estimar o custo e o tempo de desenvolvimento de um sistema.

d) assegurar que os clientes, os usuários e os desenvolvedores tenham um entendimento comum da organização na qual um sistema será implantado.

e) entender a estrutura e a dinâmica da organização na qual um sistema será implantado.

### Comentários:



(a) Errada. A arquitetura do sistema só é esboçada após os requisitos terem sido colhidos com os stakeholders; (b) Errada. A adaptação e configuração do sistema são uma coisa maior que em parte utiliza as especificidades do sistema e se baseia nos requisitos colhidos do sistema, contudo isso não é objetivo da engenharia de requisitos; (c) Correta. Custo e Tempo são uma das preocupações principais de quem desenvolve e necessita de software, e os requisitos ajudam a fornecer uma boa base para essa estimativa; (d) Errada. Pessoal, entender como a organização não é um objetivo da disciplina de requisitos; (e) Errada. Esse talvez seja um dos objetivos da disciplina de análise de negócio.

**Gabarito:** Letra C

**47. (IBFC / MGS – 2015)** A definição: “descrevem as funcionalidades que se espera que o sistema disponibilize, de uma forma completa e consistente. É aquilo que o usuário espera que o sistema ofereça, atendendo aos propósitos para qual o sistema será desenvolvido.”, corresponde tipicamente aos:

- a) Requisitos Funcionais.
- b) Requisitos Externos.
- c) Requisitos não-Funcionais.
- d) Requisitos da Aplicação.

#### Comentários:

*Funcionalidades que se espera que o sistema disponibiliza?* Já podemos dizer que se trata de requisitos funcionais.

**Gabarito:** Letra A

**48. CESGRANRIO / IBGE – 2014)** Solicitado para fazer o levantamento dos requisitos para um novo software a ser desenvolvido, um analista de sistemas identificou a necessidade de descobrir todos aqueles que se beneficiariam de forma direta ou indireta do sistema a ser desenvolvido.

Essas pessoas são conhecidas como:

- a) clientes
- b) partes interessada
- c) patrocinadores
- d) usuários
- e) usuários finais

#### Comentários:





Pessoal, só atentem para o fato de que o conceito de Partes Interessadas (Stakeholders) é mais amplo que o de Patrocinadores, uma vez que os Patrocinadores são envolvidos que tem benefícios diretos e influenciam diretamente o projeto, já as Partes Interessadas consideram mesmo aqueles que não tem qualquer influência sobre o projeto, mas mesmo de forma indireta são beneficiados/prejudicados.

**Gabarito:** Letra B

**49.(CESGRANRIO / IBGE – 2014)** Solicitado para fazer o levantamento dos requisitos para um novo software a ser desenvolvido, um analista de sistemas identificou a necessidade de descobrir todos aqueles que se beneficiariam de forma direta ou indireta do sistema a ser desenvolvido. Essas pessoas são conhecidas como:

- a) clientes
- b) partes interessadas
- c) patrocinadores
- d) usuários
- e) usuários finais.

#### **Comentários:**

(a) Errado. Refere-se às pessoas ou organizações que encomendam o desenvolvimento do software, podendo ser beneficiárias diretas ou indiretas, mas este termo não abrange todas as categorias de pessoas afetadas pelo sistema;

(b) Correto. Este termo é usado para descrever todos os indivíduos ou grupos que têm interesse no sucesso do projeto. Isso inclui não apenas os usuários finais, mas também clientes, patrocinadores, equipe de desenvolvimento, e qualquer pessoa ou entidade afetada pelo projeto. Portanto, é o termo mais abrangente que engloba tanto os beneficiários diretos quanto indiretos do sistema;

(c) Errado. São aqueles que financiam o projeto e têm interesse em seu sucesso, embora possam não ser usuários diretos do sistema;

(d) Errado. Embora os usuários se beneficiem diretamente do sistema, este termo é mais restrito e se refere especificamente àqueles que operam ou interagem com o software;

(e) Errado. Similar a "usuários", refere-se àqueles que utilizarão o sistema em seu estado final, mas não inclui todas as outras categorias de pessoas que podem se beneficiar indiretamente.

**Gabarito:** Letra B



**50. (CESGRANRIO / Petroquímica SUAPE – 2012)** Na modelagem de sistemas, há um conjunto de atividades denominado Engenharia de Requisitos. Nesse contexto, o propósito da validação de requisitos é:

- a) assegurar que os requisitos menos arriscados sejam considerados inicialmente no desenvolvimento.
- b) assegurar que o produto de software sendo especificado está em conformidade com as necessidades dos usuários.
- c) aumentar o nível de abstração na comunicação entre especialistas do domínio e engenheiros de requisitos.
- d) definir formas de rastrear os requisitos a partir dos correspondentes artefatos de modelagem e de código eventualmente existentes.
- e) maximizar o reuso de requisitos voláteis identificados em outros sistemas.

#### Comentários:

- (a) Errado. Embora a gestão de riscos seja importante em projetos de software, este não é o propósito principal da validação de requisitos;
- (b) Correto. A validação de requisitos confirma que os requisitos documentados estão alinhados com as necessidades e expectativas dos usuários e das partes interessadas, assegurando que o desenvolvimento do software vá na direção correta;
- (c) Errado. Aumentar a abstração pode ser útil na fase de elicitação ou análise de requisitos, mas não descreve o propósito da validação de requisitos;
- (d) Errado. A rastreabilidade de requisitos é importante para garantir que todos os requisitos sejam adequadamente implementados e testados, mas não é o principal objetivo da validação de requisitos;
- (e) Errado. Embora o reuso de requisitos possa ser uma prática eficiente, o foco da validação de requisitos não é sobre o reuso, mas sobre a confirmação de que os requisitos atuais refletem as necessidades do projeto em questão.

**Gabarito:** Letra B

**51. (CESGRANRIO / Petroquímica SUAPE – 2012)** Diversas técnicas são utilizadas na modelagem de sistemas de software para especificar requisitos. Duas dessas técnicas são: a com base em



casos de uso (use cases) e a com base em histórias de usuário (user stories). Essas técnicas, aplicáveis a um sistema de software, são recomendadas para:

- a) descrever a decomposição funcional recursiva desse sistema.
- b) descrever aspectos de implementação e da interface gráfica desse sistema.
- c) descrever funcionalidades desse sistema, na perspectiva das necessidades dos usuários.
- d) descrever requisitos não funcionais (tais como segurança e desempenho) desse sistema.
- e) eliminar os requisitos voláteis identificados para esse sistema.

### Comentários:

(a) Errado. Esta opção refere-se a uma abordagem mais técnica e estruturada de análise de sistemas, que não é o foco principal de casos de uso e histórias de usuário.

(b) Errado. Casos de uso e histórias de usuário focam mais nas interações e objetivos do usuário do que nos detalhes técnicos específicos de implementação ou design da interface gráfica.

(c) Correto. Esta é a principal finalidade de ambas as técnicas. Elas são usadas para entender e documentar o que os usuários precisam e como eles interagem com o sistema para atender essas necessidades.

(d) Errado. Embora casos de uso e histórias de usuário possam ajudar a identificar alguns requisitos não funcionais, eles são primariamente focados em requisitos funcionais.

(e) Errado. A eliminação de requisitos voláteis não é o foco dessas técnicas; elas são utilizadas para capturar e especificar requisitos, não para eliminá-los.

**Gabarito:** Letra C

**52. (CESGRANRIO / CAIXA – 2012)** Um dos objetivos das revisões técnicas formais de software é:

- a) realizar uma única reunião ao final do projeto para avaliar se o software foi bem construído.
- b) realizar reuniões com os clientes para descobrir o que deve ser feito.
- c) documentar os requisitos elicitados.
- d) garantir que o software não possui erros.
- e) garantir que o software atende aos requisitos especificados.

### Comentários:

(a) Errado. Esta abordagem não reflete o propósito das revisões técnicas formais, que ocorrem em várias fases do desenvolvimento, não apenas ao final.



b) Errado. Enquanto a interação com os clientes é crucial, isso é mais relevante para a fase de elicitacão de requisitos do que para as revisões tcnicas formais, que focam na avaliaão de artefatos de software j desenvolvidos.

(c) Errado. Documentar requisitos  uma etapa importante no processo de desenvolvimento de software, mas no  o objetivo principal das revisões tcnicas formais, que so mais sobre avaliaão do que sobre documentaão.

(d) Errado. Enquanto identificar e corrigir erros  um dos objetivos das revisões tcnicas,  praticamente impossvel garantir que o software esteja completamente livre de erros apenas com revisões tcnicas formais.

(e) Correto. Este  o objetivo principal das revisões tcnicas formais. Ao avaliar diferentes artefatos de software contra os requisitos especificados, as revisões ajudam a assegurar que o software desenvolvido esteja alinhado com o que foi solicitado pelos clientes e partes interessadas.

**Gabarito:** Letra E

**53. (CESGRANRIO / Petrobras – 2012)** Em um projeto de software, o documento de requisitos de software  utilizado por vrios usurios, cada um buscando no documento uma ou mais aplicaes. Os Engenheiros de Sistemas buscam, principalmente,

- a) desenvolver testes de validaão para o sistema
- b) entender qual sistema deve ser desenvolvido.
- c) entender o sistema e as relaes entre suas partes
- d) especificar os requisitos do sistema.
- e) planejar o processo de desenvolvimento do sistema.

#### Comentrios:

(a) Errado, isso  funo do Engenheiro de Testes de Sistema; (b) Correto, isso  funo do Engenheiro de Sistema; (c) Errado, isso  funo do Engenheiro de Manuteno de Sistema; (d) Errado, isso  funo dos Clientes de Sistema; (e) Errado, isso  funo dos Gerentes.

**Gabarito:** Letra B

**54. (CESGRANRIO / Petrobras – 2012)** Ao determinar os requisitos de um projeto,  necessrio levantar os requisitos funcionais e os no funcionais. Os requisitos no funcionais:

- a) definem com detalhes exatamente o que deve ser implementado.
- b) definem explicitamente as funes que o sistema no deve executar.
- c) indicam os servios que o sistema deve prestar.
- d) representam restries aos servios oferecidos pelo sistema.



e) são descrições de que serviços o sistema deve fornecer aos usuários.

### Comentários:

- (a) Errado. Esta descrição se aplica mais aos requisitos funcionais, que detalham as funcionalidades e operações do sistema.
- (b) Errado. Embora possa ser útil especificar o que um sistema não deve fazer, isso não é o foco principal dos requisitos não funcionais.
- (c) Errado. Esta descrição também é mais adequada para os requisitos funcionais, que se referem aos serviços e funções do sistema.
- (d) Correto. Eles realmente estabelecem parâmetros de qualidade, restrições operacionais, padrões de desempenho e outras condições sob as quais o sistema deve operar.
- (e) Errado. Mais uma vez, esta definição está mais alinhada com os requisitos funcionais, não com os não funcionais.

**Gabarito:** Letra D

**55. (CESGRANRIO / LIQUIGÁS – 2012)** O processo que determina se as especificações dos requisitos são consistentes com suas definições é chamado de Validação dos Requisitos. Dentre as técnicas automatizadas usadas nesse processo, citam-se os(as):

- a) cenários
- b) revisões
- c) entrevistas
- d) prototipações
- e) provas matemáticas

### Comentários:

Ora! Tanto Revisões de Requisitos quanto Prototipação são técnicas que podem ser automatizadas, logo a questão deveria ser anulada.

**Gabarito:** Letra D

**56. (CESGRANRIO / TRANSPETRO – 2012)** Conversando sobre o sistema de estoque, o usuário informou ao analista que tal sistema deverá ter interface gráfica e gerar, todos os dias, a lista de produtos perecíveis no prazo de validade. Essas informações dadas pelo usuário, referem-se, respectivamente, às seguintes definições:



- a) característica de qualidade e requisito não funcional
- b) requisito funcional e requisito de eficiência
- c) requisito não funcional e requisito funcional
- d) requisito funcional e característica de qualidade
- e) requisito de usabilidade e requisito de produto

### Comentários:

(a) Errado. Esta alternativa mistura termos de forma que não corresponde diretamente à informação fornecida; (b) Errado. Esta alternativa inverte as categorias dos requisitos informados; (c) Correto. Esta alternativa realmente identifica que ter uma interface gráfica é um requisito não funcional (relacionado à usabilidade) e que gerar a lista de produtos perecíveis é um requisito funcional; (d) Errado. Esta alternativa não classifica corretamente a primeira informação (interface gráfica) como um requisito não funcional; (e) Errado. Esta alternativa tenta categorizar as informações, mas não usa a terminologia padrão de "requisito funcional" e "requisito não funcional" de maneira clara e direta.

**Gabarito:** Letra C

**57. (CESGRANRIO / CHEST – 2012)** O processo de engenharia de requisitos engloba todas as atividades necessárias para criar e manter um documento de requisitos do sistema e compreender os elementos de negócio que serão atendidos pelo software a ser desenvolvido, pertencendo a uma sequência lógica de atividades que culminam em um documento de requisitos correto que inclui todas as necessidades do cliente. O primeiro passo necessário no processo de criação desse documento de requisitos deve ser a(o):

- a) análise de viabilidade
- b) prototipagem do sistema
- c) geração de casos de teste
- d) design da aplicação
- e) documento de gerenciamento de mudanças

### Comentários:

(a) Correto. Este é um passo crítico no início do processo de engenharia de requisitos. A análise de viabilidade ajuda a determinar se o projeto é tecnicamente e economicamente viável, se as tecnologias necessárias estão disponíveis e se o projeto pode ser completado dentro das restrições de tempo e orçamento. É uma etapa fundamental antes de aprofundar nos requisitos específicos, pois garante que o projeto faz sentido para ser desenvolvido;

(b) Errado. A prototipagem não é o primeiro passo no processo de engenharia de requisitos. Em geral, é utilizada após a identificação inicial dos requisitos para refinar e validar esses requisitos;



- (c) Errado. Embora a criação de casos de teste seja importante para a validação dos requisitos, ela ocorre depois que os requisitos são definidos e documentados;
- (d) Errado. O design da aplicação não é o primeiro passo, pois depende de um entendimento claro dos requisitos do sistema;
- (e) Errado. O gerenciamento de mudanças não é o primeiro passo no processo de engenharia de requisitos – é uma atividade contínua que começa depois que os requisitos iniciais são definidos.

**Gabarito:** Letra A

**58. (CESGRANRIO / BR DISTRIBUIDORA – 2011)** Ao determinar um conjunto inicial de requisitos funcionais para um sistema Web de controle de pedidos feitos por postos de revenda de combustível, um analista inexperiente de uma empresa distribuidora chegou aos seguintes requisitos:

- I - O sistema deverá permitir ao representante do posto solicitar uma entrega de combustíveis.  
II - O sistema deverá utilizar JAVA.  
III - O sistema deverá suportar pedidos em português e espanhol.

É(São) requisito(s) funcional(is) APENAS o(s) descrito(s) em

- a) I  
b) II  
c) III  
d) I e II  
e) II e III

#### **Comentários:**

- (I) Correto. Este é claramente um requisito funcional, pois descreve uma função específica que o sistema deve oferecer aos usuários (representantes dos postos);
- (II) Errado. Este requisito especifica uma tecnologia ou plataforma de desenvolvimento para o sistema, o que é característico de um requisito não funcional;
- (III) Errado. Trata-se de um requisito não funcional relacionado à localização e internacionalização do sistema.

**Gabarito:** Letra A





**59. (CESGRANRIO / FINEP – 2011)** Uma equipe de analistas está entrevistando gerentes de área para levantar os requisitos do novo sistema de reservas de uma companhia aérea. Considere as afirmativas sobre os requisitos levantados.

I - O usuário poderá fornecer um roteiro com múltiplos pontos de parada.

II - O total a ser pago deverá ser expresso na moeda escolhida pelo usuário.

III - As trocas de informações com os sistemas das empresas coligadas (hotéis, locadora de veículos, etc.) são feitas através de Web Services.

É(São) requisito(s) funcional(ais) o que é apresentado em.

- a) I, apenas.
- b) III, apenas.
- c) I e II, apenas.
- d) II e III, apenas.
- e) I, II e III.

#### Comentários:

(I) Correto. Este é um requisito funcional, pois descreve uma ação específica que o usuário pode realizar no sistema, ou seja, fornecer um roteiro que inclui múltiplas paradas;

(II) Correto. Este também é um requisito funcional, pois especifica que o sistema deve ser capaz de calcular e apresentar o total a ser pago em uma moeda escolhida pelo usuário;

(III) Errado. Este é um requisito não funcional que descreve um mecanismo de integração ou a tecnologia usada para a comunicação entre sistemas. Embora a integração em si possa ser vista como uma função, a especificação de usar Web Services é sobre o como, não o que o sistema deve fazer.

**Gabarito:** Letra C

**60. (CESGRANRIO / Petrobras – 2011)** Na Engenharia de Software, os requisitos que descrevem o comportamento externo do sistema, estabelecendo uma descrição detalhada das funções, dos serviços e das restrições operacionais do referido sistema, são os requisitos:

- a) funcionais
- b) externos
- c) de sistema
- d) do usuário
- e) não funcionais

#### Comentários:



(a) Errado. Requisitos Funcionais são ações ou funcionalidades que o sistema deve fornecer para atingir seus objetivos; (b) Errado. Requisitos Externos abrangem todos os requisitos derivados de fatores externos ao sistema e seu processo de desenvolvimento; (c) Correto. Requisitos de Sistema são descrições detalhadas sobre as funções, operações e restrições de sistema que definem exatamente o que deve ser implementado; (d) Requisitos do Usuário são descrições de alto nível das necessidades dos usuários ou dos objetivos que o sistema deve atender, mas são menos detalhados do que os requisitos funcionais em termos de descrição específica das funcionalidades. (e) Errado. Requisitos Não Funcionais especificam critérios que podem ser usados para julgar a operação de um sistema, em contraste com as funcionalidades específicas. Eles incluem aspectos como desempenho, segurança, usabilidade e compatibilidade.

**Gabarito:** Letra C

**61. (CESGRANRIO / Petrobras – 2010)** Um analista de sistemas levantou a lista de requisitos de um sistema de contas a pagar (SCP).

R<sub>1</sub> = O contador cadastra no SCP uma ou várias contas.

R<sub>2</sub> = Diariamente o contador visualiza na tela as contas que devem ser pagas no dia.

R<sub>3</sub> = Um relatório é gerado contendo os pagamentos realizados em um dado mês.

R<sub>4</sub> = O contador deve poder acessar o SCP através de um navegador Internet padrão.

R<sub>5</sub> = O SCP deve estar disponível durante o horário de expediente.

Como são classificados estes requisitos?

a) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>, R<sub>4</sub>, R<sub>5</sub>  
Requisitos Não Funcionais - Nenhum

b) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>  
Requisitos Não Funcionais - R<sub>3</sub>, R<sub>4</sub>, R<sub>5</sub>

c) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>  
Requisitos Não Funcionais - R<sub>4</sub>, R<sub>5</sub>

d) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>, R<sub>4</sub>  
Requisitos Não Funcionais - R<sub>5</sub>

e) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>, R<sub>4</sub>  
Requisitos Não Funcionais - R<sub>3</sub>, R<sub>5</sub>

### Comentários:

- R<sub>1</sub>: Requisito funcional, pois descreve uma função específica que o sistema deve oferecer.



- R2: Requisito funcional, relacionado à função de visualizar contas a pagar.
- R3: Requisito funcional, especifica a geração de relatórios pelo sistema, uma função específica.
- R4: Requisito não funcional, pois define um meio de acesso ao sistema (usabilidade e interoperabilidade), não uma função específica do sistema.
- R5: Requisito não funcional, relacionado à disponibilidade do sistema.

Logo, os requisitos são classificados como:

- Requisitos Funcionais: R1, R2, R3 – descrevem funções específicas que o sistema deve ser capaz de executar.
- Requisitos Não Funcionais: R4, R5 – descrevem como o sistema deve ser em termos de acesso e disponibilidade, não o que o sistema faz em termos de funcionalidades específicas.

**Gabarito:** Letra C

**62.(CESGRANRIO / ELETROBRAS – 2010)** Uma equipe de desenvolvimento de sistemas foi contratada para confeccionar o software de controle de voo de uma nova aeronave. Sabendo-se que esse assunto é novo para os membros da equipe, a primeira ação a ser tomada, no contexto de levantamento de requisitos, é

- a) determinar as necessidades iniciais do cliente pela aplicação de questionários.
- b) realizar o modelo de negócios, usando diagramas UML para descrever os processos de negócio do cliente.
- c) Aumentar o máximo possível o nível de compreensão da equipe sobre o campo de aplicação.
- d) criar rapidamente um protótipo descartável do produto, com base nas entrevistas preliminares.
- e) verificar a viabilidade das especificações não funcionais exigidas pelo cliente.

### Comentários:

(a) Errado. Embora a coleta de necessidades iniciais seja importante, aplicar questionários sem uma compreensão prévia do domínio pode levar a perguntas mal formuladas ou à falta de capacidade para interpretar corretamente as respostas.

(b) Errado. Modelar os processos de negócio é uma etapa valiosa, mas pressupõe que a equipe já tenha um entendimento razoável do domínio de aplicação, o que pode não ser o caso aqui.



(c) Correto. Esta é a ação mais crítica e a primeira que deve ser tomada quando a equipe está se aventurando em um domínio desconhecido. Uma compreensão sólida do campo de aplicação é fundamental para todas as etapas subsequentes do processo de levantamento de requisitos.

(d) Errado. A prototipagem é uma técnica útil para refinar requisitos e coletar feedback, mas fazer isso prematuramente, antes de entender o domínio, pode levar a suposições incorretas e desperdício de recursos.

(e) Errado. A avaliação da viabilidade das especificações não funcionais é importante, mas não deve ser o primeiro passo antes de adquirir uma compreensão adequada do domínio.

**Gabarito:** Letra C

**63. (CESGRANRIO / ELETROBRAS – 2010)** Para superar as dificuldades encontradas na execução do levantamento de requisitos de sistemas, uma empresa estuda as vantagens e as desvantagens de diferentes técnicas. Qual a técnica de levantamento de requisitos, baseada na observação, em que o analista se insere no ambiente de trabalho no qual o sistema será utilizado, para compreender a política organizacional e a cultura de trabalho, com o objetivo de familiarizar-se com o negócio e sua história?

- a) Workshop.
- b) JAD.
- c) Entrevista.
- d) Etnografia.
- e) Brainstorming.

#### **Comentários:**

(a) Errado. Um workshop envolve a reunião de stakeholders para discutir requisitos, mas não necessariamente inclui a observação direta do ambiente de trabalho pelos analistas.

(b) Errado. Uma técnica que envolve sessões colaborativas com usuários e desenvolvedores para definir requisitos, mas, assim como workshops, não se baseia na observação do ambiente de trabalho.

(c) Errado. Consiste em conversas diretas com os usuários e stakeholders para coletar requisitos. Embora possa fornecer insights valiosos, não envolve a observação direta e contínua do ambiente de trabalho.

(d) Correto. Corresponde à descrição fornecida, focando na observação direta do ambiente de trabalho.



(e) Errado. Uma técnica criativa para gerar ideias e identificar requisitos em grupo. Não envolve observação direta do ambiente de trabalho.

**Gabarito:** Letra D

**64.(CESGRANRIO / ELETROBRAS – 2010)** Uma fábrica de software recomenda que a documentação de especificação funcional de um sistema deve ser clara para o cliente e detalhada para o desenvolvedor, estabelecendo um contrato entre eles. Documentos de especificação funcional têm como característica:

- a) apresentar os diagramas de interação relacionados aos requisitos contidos na especificação.
- b) conter os requisitos não funcionais pertinentes ao problema a ser resolvido.
- c) conter instruções detalhadas sobre o que o sistema deve fazer e como ele deve ser implementado.
- d) ser descrito em uma linguagem matemática formal, facilitando o entendimento do cliente que, geralmente, não é um especialista técnico.
- e) definir os recursos responsáveis por implementar as funcionalidades.

#### **Comentários:**

(a) Errado. Os diagramas de interação podem fazer parte da especificação funcional para ajudar a visualizar como os diferentes componentes do sistema interagem para realizar uma função, mas essa não é uma característica definidora dos documentos de especificação funcional.

(b) Correto. Embora os requisitos não funcionais sejam importantes para o desenvolvimento do sistema, eles não são o foco principal de um documento de especificação funcional, que se concentra em descrever o que o sistema deve fazer (requisitos funcionais).

(c) Errado. A parte que diz "*como ele deve ser implementado*" está errada, visto que detalhes de implementação específicos geralmente são deixados para a fase de design técnico, não para a especificação funcional.

(d) Errado. A utilização de uma linguagem matemática formal é mais comum em especificações formais de sistemas e não necessariamente facilita o entendimento do cliente, especialmente se o cliente não for um especialista técnico.

(e) Errado. A atribuição de recursos para implementar funcionalidades específicas geralmente ocorre durante o planejamento do projeto e a gestão de tarefas, não na especificação funcional.

**Gabarito:** Letra B



**65. (CESGRANRIO / ELETROBRAS – 2010)** As técnicas de identificação de requisitos de sistemas possuem características apropriadas a cada situação. Nesse contexto, analise as afirmações sobre as técnicas a seguir, considerando que a abordagem baseada em

- I - Workshop de Requisitos utiliza momentos de descontração como forma de dinamizar o trabalho em equipe;
- II - Cenários utiliza exemplos práticos descritivos do comportamento de um sistema;
- III - Entrevistas e Questionários mostra-se inadequada na fase inicial de obtenção de dados.

Está correto o que se afirma em.

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

#### Comentários:

(I) Correto. Workshops de requisitos são sessões colaborativas que envolvem stakeholders para discutir e definir os requisitos de um sistema. Pode incluir momentos de descontração para dinamizar o trabalho em equipe, apesar de o foco principal estar em facilitar a comunicação eficaz, o entendimento mútuo dos requisitos e a colaboração;

(II) Correto. Cenários são descrições detalhadas de como os usuários podem interagir com o sistema em situações específicas. Eles ajudam a identificar requisitos funcionais e não funcionais ao fornecer exemplos práticos do comportamento esperado do sistema;

(III) Errado. Entrevistas e questionários são frequentemente utilizados exatamente na fase inicial de um projeto para obter dados, entender as necessidades dos stakeholders e começar a identificar os requisitos. Eles são métodos eficazes para coletar informações diretamente das partes interessadas.

**Gabarito:** Letra B

**66. (CESGRANRIO / Petrobras – 2010)** No contexto da engenharia de software, o processo conhecido como engenharia de requisitos permite ao engenheiro de software

- a) eliminar a volatilidade dos requisitos que foram levantados para o produto de software.
- b) postergar a definição do escopo inicial do sistema para etapas posteriores do desenvolvimento.



- c) realizar a Implantação da Função de Qualidade (IFQ), que corresponde a maximizar a qualidade do processo de software utilizado no desenvolvimento do produto de software já implantado.
- d) remover as dependências entre os requisitos do sistema e as regras de negócio.
- e) usar o modelo de casos de uso para especificar os requisitos funcionais do produto de software.

### Comentários:

(a) Errado. Embora a engenharia de requisitos busque minimizar mudanças nos requisitos ao melhorar a compreensão e o acordo sobre estes, eliminar completamente a volatilidade dos requisitos é impraticável, pois mudanças podem ocorrer devido a diversas razões, como novas descobertas durante o desenvolvimento ou mudanças nas necessidades de negócio.

(b) Errado. Na verdade, a engenharia de requisitos visa definir o escopo inicial do sistema o mais cedo possível para guiar o desenvolvimento subsequente, ao invés de postergar essa definição.

(c) Errado. Esta descrição parece misturar conceitos e não reflete diretamente um objetivo da engenharia de requisitos. A engenharia de requisitos foca mais em requisitos do produto do que na qualidade do processo de desenvolvimento.

(d) Errado. Na verdade, a engenharia de requisitos muitas vezes envolve alinhar os requisitos do sistema às regras de negócio, ao invés de remover suas dependências. As regras de negócio são fundamentais para definir os requisitos de um sistema.

(e) Correto. Os casos de uso são uma técnica eficaz para especificar os requisitos funcionais, descrevendo as interações entre os usuários (atores) e o sistema para realizar objetivos específicos. Eles ajudam a capturar, em termos claros, o que o sistema deve fazer do ponto de vista do usuário.

**Gabarito:** Letra E

**67.(CESGRANRIO / IBGE – 2010)** Com o objetivo de minimizar os problemas enfrentados e melhorar o processo de engenharia de requisitos, um engenheiro de requisitos decidiu elencar uma série de medidas que poderá empregar em seus futuros projetos, tais como:

I - aplicar a técnica de IFQ (Implantação da Função de Qualidade) que permite coletar os requisitos excitantes, os quais refletem características que vão além das expectativas do cliente e mostram ser muito satisfatórios quando presentes;

II - utilizar tabelas de rastreamento que relacionam os requisitos identificados a um ou mais aspectos do sistema;





III - utilizar casos de uso para fazer uma coleta iterativa de requisitos, uma vez que o processo de levantamento de requisitos é uma atividade evolutiva.

Está(ão) correta(s) a(s) medida(s)

- a) I, apenas.
- b) II, apenas.
- c) III, apenas.
- d) I e II, apenas.
- e) I, II e III.

### Comentários:

(I) Correto. Esta técnica é usada para transformar as necessidades dos clientes em requisitos específicos de produtos ou serviços, priorizando-os. O termo "requisitos excitantes" se refere à teoria de Kano, que classifica os requisitos em básicos, de desempenho e excitantes. Os requisitos excitantes são aqueles que podem surpreender positivamente o cliente, oferecendo características inesperadas que geram alta satisfação.

(II) Correto. As tabelas de rastreamento são uma ferramenta essencial na engenharia de requisitos para relacionar os requisitos identificados com outros elementos do sistema, como casos de teste, componentes de software, e requisitos de nível superior. Elas ajudam a garantir a cobertura dos requisitos e facilitam a gestão de mudanças.

(III) Correto. Casos de uso são uma técnica eficaz para descrever as interações entre os usuários (atores) e o sistema, capturando os requisitos funcionais do sistema de uma maneira que é compreensível tanto para stakeholders técnicos quanto não técnicos. A natureza iterativa do levantamento de requisitos é reconhecida como uma prática importante, permitindo refinamentos contínuos à medida que mais informações são coletadas e compreendidas.

**Gabarito:** Letra E

**68. (ESAF / CVM – 2010)** Assinale a opção correta.

- a) Gestão de requisitos preocupa-se com a documentação, atualização e controle de stakeholders envolvidos na fase de identificação da demanda.
- b) Engenharia de requisitos compreende: identificar, analisar, especificar e definir as necessidades de negócio que um aplicativo deve prover para solução do problema levantado.
- c) Engenharia de requisitos compreende: planejar, especificar e desenvolver as necessidades de negócio que um aplicativo deve prover para minimização dos problemas levantados.



- d) Engenharia de requisitos compreende: identificar, analisar, programar e testar os programas das necessidades de solução de problemas que um negócio deve prover para satisfazer usuários.
- e) Gestão de requisitos preocupa-se com a documentação, direcionamento, controle de definição e acesso aos requisitos levantados na fase de planejamento de escopo.

### Comentários:

(a) Errado, documentação, atualização e controle de stakeholders não é gestão de requisitos; (b) Correto, trata das necessidades para solucionar um problema; (c) Errado, na engenharia de requisitos não se desenvolve as necessidades de negócio, entre outros erros; (d) Errado, programar não é uma das atividades, entre outros erros; (e) Errado, planejamento de escopo não é foco da engenharia de requisitos.

**Gabarito:** Letra B

**69. (ESAF / MPOG – 2010)** As áreas de esforços da Análise de Requisitos são:

- a) reconhecimento dos objetivos, avaliação e controle, modelagem, estruturação e revisão.
- b) reconhecimento do problema, avaliação e síntese, modelagem, especificação e revisão.
- c) reengenharia, planejamento, avaliação e controle, modelagem e conclusão.
- d) reconhecimento do problema, análise e síntese, reengenharia, especificação e análise de resultados.
- e) reconhecimento do problema, modelagem, especificação de entidades, estruturação e revisão.

### Comentários:

Essa questão foi retirada do livro do Denis Alcides Rezendo, que afirma que, a partir do relato e necessidades, a análise de requisitos possibilita que o Engenheiro de Software especifique as funções, o desempenho, interfaces, restrições, etc do software. Proporciona avaliar a qualidade de atendimento e satisfação, podendo ser dividido em cinco áreas de esforço: reconhecimento do problema, avaliação e síntese, modelagem, especificação e revisão.

**Gabarito:** Letra B

**70. (CESGRANRIO / TJ-RO – 2008)** Considere os quatro requisitos registrados em um projeto de uma aplicação para a Internet apresentados a seguir.



I - O tempo de resposta máximo do sistema a qualquer ação do usuário deve ser de 5s.

II - Clientes que tenham pago as últimas cinco compras à vista têm direito a um desconto não cumulativo de 10% na próxima compra.

III - A interface com o usuário deve ser organizada em abas e menus.

IV - Se o produto possuir uma quantidade máxima permitida por compra, esse limite deve ser imposto pelo sistema durante uma compra.

São tipicamente classificados como requisitos funcionais APENAS os requisitos.

- a) I e II
- b) I e III
- c) II e III
- d) II e IV
- e) III e IV

#### Comentários:

(I) Errado. Este é um requisito não funcional, pois refere-se ao desempenho do sistema, especificando um critério de operação (tempo de resposta) em vez de uma funcionalidade específica.

(II) Correto. Este é um requisito funcional, pois descreve uma funcionalidade específica que o sistema deve implementar: calcular e aplicar um desconto com base no histórico de compras do cliente.

(III) Errado. Este é um requisito não funcional relacionado à usabilidade, especificando aspectos da interface do usuário, mas não descreve diretamente uma funcionalidade do sistema.

(IV) Correto. Este também é um requisito funcional, pois especifica uma regra de negócio que o sistema deve cumprir ao realizar uma compra, ou seja, impor um limite na quantidade de produto que pode ser comprado.

**Gabarito:** Letra D

**71. (CESGRANRIO / Petrobras – 2008)** Analise as afirmativas a seguir, sobre requisitos em projetos de software.

I - O rastreamento de requisitos é de grande importância para conduzir análises de impacto quando há mudanças em requisitos.



II - O acrônimo FURPS+ se refere aos requisitos não funcionais das categorias de Feasibility, Usability, Reliability, Performance e Supportability.

III - Um requisito pode conter, além da especificação, atributos que sirvam ao seu gerenciamento.

IV - Casos de uso são descrições da interação entre um ator e o sistema e, portanto, especificam apenas requisitos funcionais.

Estão corretas APENAS as afirmativas.

- a) I e II
- b) I e III
- c) II e III
- d) II e IV
- e) III e IV

### Comentários:

(I) Correto. O rastreamento de requisitos permite identificar e gerenciar as relações entre os requisitos e outros elementos do projeto, facilitando a análise de impacto de mudanças nos requisitos.

(II) Errado. Esta afirmativa está incorreta. O acrônimo FURPS+ de fato se refere a requisitos não funcionais, mas as categorias são: Functionality, Usability, Reliability, Performance, Supportability (mais as categorias adicionais indicadas pelo "+", como Design Constraints, Implementation Requirements, Interface Requirements, Physical Requirements, etc.). "Feasibility" (Viabilidade) não é parte do acrônimo FURPS+.

(III) Correto. Além da descrição do que o requisito deve fazer ou cumprir, ele pode incluir atributos para seu gerenciamento, como prioridade, fonte, status e rastreabilidade.

(IV) Errado. Casos de uso descrevem como os usuários (atores) interagem com o sistema para alcançar determinados objetivos, especificando assim requisitos funcionais do sistema.

**Gabarito:** Letra B

**72. (CESGRANRIO / TJ-RO – 2008)** Durante as atividades de Requisitos em um projeto de desenvolvimento de software, são realizadas entrevistas com clientes (usuários e stakeholders, no papel de entrevistados) com o objetivo de levantar suas necessidades e validar as características propostas para o software a ser desenvolvido. Os analistas, no papel de entrevistadores, em geral utilizam dois tipos de perguntas durante as entrevistas: perguntas



livres de contexto e perguntas no contexto da solução. Sobre o tema, assinale a afirmativa correta.

a) Perguntas livres de contexto proporcionam ao analista um entendimento do problema a ser resolvido pelo sistema sem influenciar o entrevistado com detalhes de uma solução que o analista já tenha em mente, baseada em experiências prévias, que pode não ser a mais adequada para o projeto.

b) Perguntas no contexto da solução devem ser feitas antes das perguntas livres de contexto, de forma que seja possível obter um entendimento inicial do escopo do projeto e testar se a solução proposta coincide com as expectativas dos clientes.

c) Perguntas no contexto da solução estão relacionadas à identificação das necessidades dos clientes, enquanto perguntas livres de contexto estão relacionadas às características do software a ser desenvolvido.

d) Um exemplo de pergunta no contexto da solução é "Quem são os usuários (do software a ser desenvolvido)?".

e) É consenso nos dias atuais que uma alternativa vantajosa às entrevistas é a distribuição de questionários aos clientes, com perguntas dos dois tipos, o que possibilita colher mais informações em menor tempo e comparar as respostas para detectar inconsistências.

### Comentários:

(a) Correto. Perguntas livres de contexto são essenciais para entender as necessidades e problemas dos usuários sem pré-conceitos ou soluções predefinidas, permitindo um levantamento mais puro e abrangente das necessidades reais.

(b) Errado. Esta abordagem não é ideal, pois introduzir soluções específicas muito cedo pode limitar a compreensão completa das necessidades dos usuários e influenciar suas respostas, limitando potencialmente a inovação e a adequação da solução.

(c) Errado. Na verdade, perguntas livres de contexto ajudam a identificar as necessidades dos clientes sem a influência de soluções específicas, enquanto perguntas no contexto da solução focam em detalhes específicos da implementação ou funcionalidades do software.

(d) Errado. Esta pergunta é mais neutra e pode ser considerada uma pergunta livre de contexto, pois busca entender o ambiente e os atores envolvidos sem se referir a uma solução específica.

(e) Errado. Embora questionários possam ser úteis para coletar dados de muitos usuários rapidamente, eles não substituem completamente as entrevistas, que permitem uma interação mais profunda e a possibilidade de esclarecer dúvidas imediatamente. Não há um consenso de que



questionários sejam uma alternativa vantajosa às entrevistas; cada método tem suas vantagens e limitações.

**Gabarito:** Letra A

**73. (CESGRANRIO / Petrobras – 2006)** Sobre a Análise e o Gerenciamento de Requisitos, é FALSO afirmar que:

- a) quanto mais tarde for identificado um problema na análise de requisitos, maior será o custo com o retrabalho.
- b) a elicitação é o processo de identificação e entendimento das necessidades e restrições dos usuários, enquanto que a especificação é o processo de formalização das necessidades e restrições dos usuários em requisitos funcionais de software.
- c) na análise de requisitos o cliente utiliza as melhores práticas de engenharia de requisitos na tarefa de descrever suas necessidades.
- d) o gerenciamento de requisitos corresponde ao conjunto de atividades que auxilia a equipe do projeto a identificar, controlar e rastrear os requisitos, bem como a fazer as alterações nos requisitos durante o projeto.
- e) o gerenciamento de requisitos implica a alteração, inclusão e/ou exclusão de requisitos ao produto de software, o que pode levar a alterações de prazos, de recursos humanos, de equipamentos e de tecnologia.

### Comentários:

- (a) Correto. Erros de requisitos identificados em fases posteriores do desenvolvimento geralmente resultam em custos significativamente maiores de correção.
- (b) Correto. Elicitação e especificação são duas etapas cruciais na engenharia de requisitos, focadas, respectivamente, na coleta e na documentação formal dos requisitos.
- (c) Errado. Embora o cliente participe do processo de análise de requisitos ao descrever suas necessidades, geralmente são os analistas de requisitos ou engenheiros de software que aplicam as melhores práticas de engenharia de requisitos para identificar, analisar e documentar essas necessidades de forma sistemática. Os clientes podem não ter conhecimento dessas práticas e dependem dos especialistas para conduzir o processo adequadamente.
- (d) Correto. O gerenciamento de requisitos é essencial para garantir que os requisitos sejam devidamente acompanhados e mantidos atualizados ao longo do projeto.



(e) Correto. Mudanças nos requisitos podem ter impactos significativos no escopo do projeto, nos custos e nos prazos.

**Gabarito:** Letra C

---

**74. (CESGRANRIO / Petrobras – 2005)** Uma classificação frequentemente utilizada para os requisitos de um sistema os classifica em funcionais, não funcionais e de domínio. Segundo a definição que os descreve corretamente, requisitos não-funcionais são:

- a) ferramentas de apoio na elaboração e acompanhamento do cronograma do projeto, como PERT e CPM, e que permitem avaliar o processo de desenvolvimento
- b) requisitos de teste utilizados para detectar problemas de lógica de programação e de modelagem de funcionalidades nas fases iniciais do processo de desenvolvimento.
- c) requisitos que definem o que o sistema deve fazer e como deve se comportar diante de determinadas entradas fornecidas pelos usuários.
- d) requisitos derivados do domínio da aplicação do sistema que refletem as características e as restrições deste domínio.
- e) requisitos que não estão diretamente relacionados com as funções específicas do sistema.

#### **Comentários:**

- (a) Errado. Esta alternativa descreve ferramentas de gerenciamento de projeto e não está relacionada à definição de requisitos não funcionais.
- (b) Errado. Esta alternativa fala sobre requisitos de teste, que são diferentes de requisitos não funcionais. Requisitos de teste estão mais relacionados à verificação e validação do sistema.
- (c) Errado. Esta descrição se aplica mais adequadamente aos requisitos funcionais, que especificam as funções que o sistema deve executar.
- (d) Errado. Requisitos de domínio são um tipo específico de requisito, mas esta alternativa não descreve os requisitos não funcionais de forma geral.
- (e) Correto. Eles especificam critérios que podem ser usados para julgar a operação de um sistema, mas não descrevem as funções específicas que o sistema deve executar.

**Gabarito:** Letra E

---





**75. (ESAF / AFRFB – 2005)** Durante a análise de requisitos, são especificados a função e o desempenho do software, bem como a sua interface com outros elementos do sistema. Nessa etapa, também, são estabelecidas as restrições de projeto, a que o software deve atender.

**Comentários:**

Pessoal, o que essa questão quis dizer? Especificar função = Requisitos Funcionais; Especificar Desempenho e Restrições = Requisitos Não-Funcionais.

---

**Gabarito:** Correto

**76. (ESAF / AFRFB – 2005 – Letra E)** Durante a especificação dos requisitos, são estabelecidos os critérios que permitirão ao desenvolvedor e ao cliente avaliar a qualidade, assim que o software for construído.

**Comentários:**

Perfeito! A partir da especificação de requisitos que se avalia a qualidade.

---

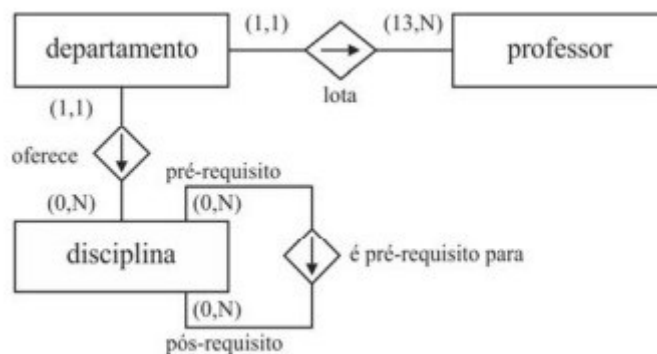
**Gabarito:** Correto



## LISTA DE QUESTÕES – CESPE

1. (CESPE / CAU-BR – 2024) Em engenharia de requisitos, o modelo conceitual é construído na fase de projeto do sistema de informação.
2. (CESPE / CAU-BR – 2024) No contexto da engenharia de requisitos e de metodologias de desenvolvimento de software, julgue o próximo item.

No diagrama entidade e relacionamento a seguir, verifica-se um exemplo clássico de autorrelacionamento para a entidade Disciplina.



3. (CESPE / MPE-GO – 2024) Em levantamento de requisitos baseado em cenários, um ator de casos de uso equivale tipicamente a um usuário que exerça papéis diferentes na operação do sistema a ser construído.
4. (CESPE / LNA – 2024) A figura precedente ilustra um processo de elicitação e análise de requisitos. Em relação a esse processo, a técnica casos de uso



a) poderia ser utilizada na Priorização e negociação de requisitos, mas somente se o sistema for orientado a objetos, pois, nesse caso, haveria a especificação de cada caso de uso.



- b) seria uma atividade do processo Descoberta de requisitos.
- c) não estaria relacionada ao referido processo, pois, como o processo é um diagrama da UML, a técnica seria utilizada somente ao final do processo para a documentação do projeto.
- d) está mais relacionada à Especificação de requisitos, uma vez que nela podem ser detalhadas as iterações do sistema.
- e) seria melhor classificada, por envolver a descrição de quem são os atores do sistema, como uma atividade da Priorização e negociação de requisitos, uma vez que envolve a definição de papéis.
5. (CESPE / ANVISA – 2024) A entrevista com o usuário, a condição de uma sessão de brainstorming e o estudo de sistemas semelhantes são exemplos de técnicas para obtenção de requisitos.
6. (CESPE / MPO – 2024) Os requisitos do sistema devem descrever apenas o comportamento externo deste, logo não abrangem a forma como o sistema deve ser projetado ou implementado.
7. (CESPE / MPO – 2024) No escopo da modelagem de requisitos, que visa criar várias representações do que o cliente deseja, os modelos baseados em cenários representam os requisitos de software sob o ponto de vista do usuário; nesse modelo, o caso de uso descreve a interação entre um ator e o software.
8. (CESPE / MPO – 2024) Os requisitos não funcionais, mesmo não tendo relação direta com os serviços específicos oferecidos pelo sistema a seus usuários, podem estar relacionados, por exemplo, à confiabilidade e ao tempo de resposta do sistema.
9. (CESPE / MPO – 2024) Descrever aquilo que o sistema não deve fazer pode ser considerado, em alguns casos, como requisito funcional.
10. (CESPE / MPO – 2024) Por meio da técnica de elicitação de requisitos, que visa permitir descobertas de requisitos funcionais, o analista de requisitos realiza imersão no ambiente de trabalho no qual o sistema será usado e, por meio de observações, extrai os requisitos.
11. (CESPE / MPO – 2024) Os processos de engenharia de requisitos podem incluir quatro atividades de alto nível: validação, especificação, estudo de viabilidade e elicitação, e, nessa última etapa, descobrem-se os requisitos.
12. (CESPE / FINEP – 2024) Em engenharia de software, a busca de conformidade aos requisitos funcionais explicitamente declarados, por meio de padrões de desenvolvimento documentados para atender as características implícitas esperadas do software, é objeto da:



- a) viabilidade.
- b) governança.
- c) usabilidade.
- d) qualidade.
- e) manutenibilidade.

13. (CESPE / MPO – 2024) Na criação das histórias de usuários, o modelo dos 3C refere-se a: cartões, conversas e confirmações.
14. (CESPE / MPO – 2024) As *story points* são a métrica para planejar o esforço na execução de projetos de software utilizando a análise por ponto de função.
15. (CESPE / TST – 2024) No processo de desenvolvimento de software ágil, uma narrativa que relata, sob a perspectiva do usuário final, como o software irá agregar valor ao negócio é denominada:
- a) sprint.
  - b) iniciativa.
  - c) história de usuário.
  - d) requisito não funcional.
  - e) épico.
16. (CESPE / CAU-BR – 2024) Ao se especificar a nova versão de um software, devem ser mapeados os requisitos não funcionais, que definem o que o sistema deve fazer para atender as expectativas do cliente.
17. (CESPE / CNPq – 2024) Após a definição dos objetivos gerais, o protótipo, nessa modelagem, atua como um mecanismo para identificar os requisitos do software.
18. (CESPE / CNPq – 2024) Um caso de uso, independentemente da sua forma, representa o software ou o sistema do ponto de vista do analista.
19. (CESPE / DATAPREV – 2023) A linguagem natural estruturada é uma forma de escrever os requisitos de sistema na qual a liberdade do elaborador de requisitos é limitada e todos os requisitos são redigidos de maneira padronizada.
20. (CESPE / SEFIN de Fortaleza-CE – 2023) Um critério de aceitação, depois de testado, significa que um requisito foi implementado corretamente.
21. (CESPE / SEFIN de Fortaleza-CE – 2023) As histórias de usuários podem ser divididas em partes menores chamadas de épicos.
22. (CESPE / SEFIN de Fortaleza-CE – 2023) A seguinte frase representa a sintaxe de uma história de usuário: como [persona], eu [quero], [para que].



- 23. (CESPE / SEFIN de Fortaleza-CE – 2023)** Um requisito funcional pode ser expresso em termos de segurança e desempenho de um software.
- 24. (CESPE / SEFIN de Fortaleza-CE – 2023)** Requisito é uma condição necessária que o sistema deverá realizar para atender a uma necessidade do usuário.
- 25. (CESPE / FUB – 2023)** Um protótipo pode demonstrar conceitos, validar opções de projeto e, ainda, ajudar na elicitação e validação de requisitos de um sistema.
- 26. (CESPE / FUB – 2023)** Etnografia é uma técnica de elicitação de requisitos que pode ser usada para compreender processos operacionais por meio da imersão no ambiente de trabalho em que o sistema será usado, porém é ineficaz em descobrir requisitos derivados da cooperação e do conhecimento das atividades das pessoas.
- 27. (CESPE / MPE-RO – 2023)** Em engenharia de requisitos, os usuários respondem quais são os objetivos do software na fase de:
- a) levantamento.
  - b) concepção.
  - c) negociação.
  - d) elaboração.
  - e) especificação.
- 28. (CESPE / DATAPREV – 2023)** Considerados uma técnica de elicitação de requisitos, os casos de uso identificam os atores envolvidos, assim como interações individuais entre o sistema e seus usuários ou, ainda, com outros sistemas.
- 29. (CESPE / DATAPREV – 2023)** Considerando-se que o processo de engenharia de requisitos seja uma espiral iterativa com três estágios — elicitação, especificação e validação de requisitos —, é correto afirmar que, nesse contexto, a atividade de prototipação é mais bem relacionada à elicitação do que à validação de requisitos.
- 30. (CESPE / DATAPREV – 2023)** Os requisitos de usuário de um sistema devem descrever os requisitos funcionais, de modo que eles sejam compreensíveis pelos usuários do sistema que não possuem conhecimento técnico detalhado, abstendo-se de inserir requisitos não funcionais.
- 31. (CESPE / TBG – 2023)** Requisito funcional especifica o que o software deve fazer, enquanto requisito não funcional especifica como o sistema deve se comportar.
- 32. (CESPE / TC-DF – 2023)** A rastreabilidade horizontal refere-se à capacidade de seguir um requisito ou uma funcionalidade de um software desde a sua origem até seus desdobramentos e implementações subsequentes.



- 33. (CESPE / BANRISUL – 2022)** Requisitos não funcionais de um sistema descrevem seu objetivo e dependem do tipo de software a ser desenvolvido, dos usuários esperados para o software e da abordagem geral adotada pela organização ao escrever os requisitos.
- 34. (CESPE / BANRISUL – 2022)** Requisitos organizacionais são requisitos de sistema amplos, derivados das políticas e dos procedimentos nas organizações do cliente e do desenvolvedor, cujas funções incluem definir como o sistema será utilizado e especificar a linguagem de programação.
- 35. (CESPE / BANRISUL – 2022)** Os requisitos do sistema devem descrever os comportamentos interno e externo do sistema, devendo-se preocupar com a forma como ele deve ser projetado ou implementado.
- 36. (CESPE / BANRISUL – 2022)** A especificação de requisitos é frequentemente composta de vários tipos de documentos e não raro abrange: visão geral; glossário; modelos do sistema; lista de requisitos funcionais e lista de requisitos não funcionais; especificação detalhada de requisitos.
- 37. (CESPE / BANRISUL – 2022)** O objetivo principal da especificação é documentar todas as necessidades dos clientes e obter um aceite quanto às entregas de produto propostas.
- 38. (CESPE / BANRISUL – 2022)** Na execução da técnica de apprenticing (aprendizado), o engenheiro de requisitos deve questionar procedimentos operacionais complexos e pouco claros do domínio do sistema que os stakeholders desejam preservar.
- 39. (CESPE / BANRISUL – 2022)** Em situações em que alguma das partes interessadas não consiga expressar de forma oral as suas necessidades com clareza, recomenda-se o emprego da técnica da etnografia para o levantamento de requisitos.
- 40. (CESPE / BANRISUL – 2022)** O levantamento de requisitos com casos de uso é muito eficaz para a eliciação de requisitos não funcionais.
- 41. (CESPE / BANRISUL – 2022)** A analogia é uma técnica pouco recomendada quando é necessário identificar requisitos novos, inovadores ou atrativos, em um ambiente cujo objetivo é encontrar soluções criativas.
- 42. (CESPE / BANRISUL – 2022)** A arqueologia é uma técnica apropriada quando se busca preservar todas as funcionalidades de um sistema legado em um novo sistema que reutilize as soluções e experiências existentes.
- 43. (CESPE / MPC-SC – 2022)** A etnografia é o processo de eliciação por meio do qual o analista de requisitos realiza uma imersão no ambiente de trabalho em que o sistema será utilizado para tornar compreensíveis os processos operacionais e auxiliar na extração dos requisitos de apoio de tais processos.



44. (CESPE / BNB – 2022) Para capturar os requisitos da interface de um sistema, os protótipos podem ser desenhados como mockups, mesmo que estes não permitam interações do usuário com a execução das funcionalidades.
45. (CESPE / BNB – 2022) Um dos critérios de boa qualidade para uma história de usuário é o denominado critério pequeno, ou seja, aquele cujo desenvolvimento da história deve representar um trabalho desenvolvido dentro de um limite de tempo de duração específica.
46. (CESPE / BNB – 2022) Em uma história de usuário, em que se deseja fazer login com a impressão digital do cliente para o seu acesso à sua conta bancária, um exemplo correto de critério de aceitação é: dado que estou realizando login com minha digital, quando eu colocar o dedo cadastrado no leitor, então consigo acessar minha conta.
47. (CESPE / BNB – 2022) No gerenciamento de requisitos, uma adequada configuração, em particular, de uma especificação tem a propriedade de ser imutável.
48. (CESPE / BANRISUL – 2022) O levantamento de requisitos com casos de uso é muito eficaz para a elicitacão de requisitos não funcionais.
49. (CESPE / BANRISUL – 2022) Em situações em que alguma das partes interessadas não consiga expressar de forma oral as suas necessidades com clareza, recomenda-se o emprego da técnica da etnografia para o levantamento de requisitos.
50. (CESPE / BANRISUL – 2022) A acessibilidade está relacionada à facilidade com que determinada informação é assimilada por pessoas com alguma deficiência.
51. (CESPE / BANRISUL – 2022) A usabilidade é um atributo de qualidade de um projeto que avalia se ele fornece os recursos que os usuários precisam.
52. (CESPE / FUNPRESP-EXE - 2022) As verificações de validade, consistência e completude são técnicas fundamentais do processo de validação de requisitos.
53. (CESPE / FUNPRESP-EXE - 2022) Dentre as técnicas existentes de elicitacão de requisitos baseadas em cenários, os casos de uso são modelos que ajudam a identificar agentes e interações do sistema.
54. (CESPE / FUNPRESP-EXE - 2022) A técnica Quality Function Deployment tem como objetivo traduzir os requisitos técnicos em requisitos do cliente.
55. (CESPE / FUNPRESP-EXE - 2022) O protótipo de interface do usuário é o produto final da técnica de prototipação da engenharia de requisitos.
56. (CESPE / FUNPRESP-EXE - 2022) Brainstorming é uma técnica utilizada para o levantamento de requisitos; para facilitar o registro, essa técnica deve ser feita por meio de questionário.





- 57. (CESPE / Petrobrás - 2022)** Ferramentas automatizadas para armazenamento de requisitos, gerenciamento de mudanças e gerenciamento de rastreabilidade são indicadas para apoio ao processo de gerenciamento de requisitos.
- 58. (CESPE / Petrobrás - 2022)** Histórias de usuário são ferramentas para a definição de escopo de produtos de software voltadas a fornecer uma análise detalhada sobre a atividade do usuário e a viabilizar a retenção de conhecimento em longo prazo.
- 59. (CESPE / Petrobrás - 2022)** Os critérios de aceitação descrevem um conjunto mínimo de requisitos que precisam ser atendidos para que valha a pena implementar uma solução específica.
- 60. (CESPE / Petrobrás - 2022)** Entrevistas e questionários são técnicas comumente usadas para obter informações relacionadas às necessidades de grupos de usuários representados por personas, que exemplificam como um usuário típico interage com um produto.
- 61. (CESPE / Petrobrás - 2022)** No contexto de storytelling, é fundamental mitigar as possibilidades de navegação por meio das interfaces e impor à experiência do usuário o sequenciamento estrito das atividades que constituem a sua história.
- 62. (CESPE / TJ-RJ - 2021)** Na engenharia de requisitos, por estar mais aderente às características dessa técnica, a etnografia é recomendada:
- a) na eliciação da forma como o fluxo dos processos deveria ser feito.
  - b) na descoberta dos requisitos organizacionais.
  - c) quando se deseja obter uma visão do funcionamento do sistema na forma prevista, independentemente das interferências de seu contexto.
  - d) na descoberta de requisitos derivados do conhecimento das atividades de outras pessoas que realizam trabalhos adjacentes ao analisado.
  - e) como uma alternativa aos casos de uso para a descoberta dos requisitos explícitos.
- 63. (CESPE / TJ-RJ - 2021)** Para os propósitos da modelagem dos requisitos com base em cenários, um suporte apropriado é o uso de
- a) diagrama de casos de uso e histórias de usuários.
  - b) diagrama de sequência e diagrama de atividades.
  - c) diagramas que representem eventos ou estados.
  - d) diagrama de classes e histórias de usuário.
  - e) modelagem com cartões CRC e casos de uso.
- 64. (CESPE / TELEBRÁS - 2021)** No âmbito da engenharia de software, o principal produto da engenharia de requisitos é o documento de especificação de requisitos de software.



- 65. (CESPE / TCE-RJ – 2021)** O gerenciamento de requisitos trata do desenvolvimento de software por meio da metodologia ágil; isso permite o isolamento entre o desenvolvedor e o usuário, já que é comum ocorrer problema de mudanças de requisitos ao longo do curso do projeto devido ao interfaceamento do usuário com o desenvolvedor.
- 66. (CESPE / TCE-RJ – 2021)** Em um processo de desenvolvimento de software, a elicitação de requisitos serve para identificar os fatos que compõem os requisitos do sistema.
- 67. (CESPE / Ministério da Economia – 2020)** Um dos princípios em que se baseia a técnica de dinâmica de grupo conhecida como brainstorm é o atraso de julgamento, que possibilita a geração de muitas ideias antes de se decidir por uma.
- 68. (CESPE / Ministério da Economia – 2020)** Os requisitos do software mudam com frequência, mas é sempre possível acomodá-los no sistema, pois o software é flexível.
- 69. (CESPE / Ministério da Economia – 2020)** Requisitos funcionais envolvem as características de confiabilidade e de desempenho de um sistema.
- 70. (CESPE / Ministério da Economia – 2020)** Elicitar requisitos não inclui somente necessidades dos usuários, mas também extrair informações que surgem de padrões organizacionais, governamentais e industriais em geral, para atender necessidades.
- 71. (CESPE / TJ-AM – 2019)** A validação dos requisitos exclui diversas considerações, entre elas, a que verifica o impacto da implementação dos requisitos identificados sobre o orçamento do sistema.
- 72. (CESPE / TJ-AM – 2019)** Na gerência de requisitos, as mudanças no documento de requisitos devem aumentar as referências a outros documentos e aprimorar a interdependência entre suas próprias seções.
- 73. (CESPE / TJ-AM – 2019)** Uma especificação de requisitos é inconsistente quando, por exemplo, em um de seus subconjuntos conste que o pagamento será feito antes do fechamento da compra e, em outro subconjunto, conste que o pagamento será feito depois do fechamento da compra.
- 74. (CESPE / TJ-AM – 2019)** Em um protótipo para validar os requisitos de um software, é admissível deixar de fora os requisitos não funcionais ou reduzir seus padrões.
- 75. (CESPE / SLU-DF – 2019)** A interoperabilidade entre um software que esteja em desenvolvimento e outros sistemas existentes é considerada um requisito funcional.
- 76. (CESPE / STM – 2018)** Requisitos de domínio são relativos ao que o sistema deve fornecer, como ele deve reagir a entradas específicas e se comportar em determinadas situações, enquanto os requisitos funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema.



- 77. (CESPE / STM – 2018)** O processo de verificação visa assegurar que o sistema atende as expectativas e necessidades do cliente por meio da utilização de técnicas de entrevista como brainstorming, grupos focais ou Delft, a partir das quais são extraídos os requisitos não funcionais.
- 78. (CESPE / CGM-PB – 2018)** A atividade de gerência de requisitos é a responsável por garantir que mudanças nos requisitos sejam feitas de maneira controlada e documentada, administrando os relacionamentos entre os requisitos e as dependências entre o documento de requisitos e os demais artefatos produzidos no processo de software.
- 79. (CESPE / ABIN – 2018)** Ao se aplicar a rastreabilidade bidirecional, é possível determinar se todos os requisitos-fonte foram completamente tratados e se todos os requisitos do produto atendem aos requisitos do cliente.
- 80. (CESPE / ABIN – 2018)** Definir e manter matriz de rastreabilidade dos requisitos permite controlar e tratar as mudanças em requisitos durante o processo de elicitação e especificação do produto.
- 81. (CESPE / ABIN – 2018)** Para que os requisitos sejam refinados e sejam gerados modelos de análise e projeto para codificação, apenas a avaliação e a aprovação por parte do cliente — mesmo após o entendimento dos requisitos — não são suficientes.
- 82. (CESPE / ABIN – 2018)** De acordo com as técnicas facilitadoras de especificação de aplicação, recomenda-se que a descrição de requisitos e regras seja feita diretamente pela equipe técnica, sem a participação do cliente.
- 83. (CESPE / ABIN – 2018)** No processo de elicitação de requisitos, há atividades relacionadas a identificação, rastreabilidade e mudanças em requisitos.
- 84. (CESPE / EBSERH – 2018)** Na especificação de requisitos, são estabelecidos uma escala de medição e os valores aceitáveis para cada requisito de usuário, tornando-o mensurável, ou seja, adicionando a ele um critério de aceitação.
- 85. (CESPE / EBSERH – 2018)** Requisitos externos são derivados de metas, políticas e procedimentos das organizações, do cliente e do desenvolvedor e incluem requisitos de processo, requisitos de implementação, restrições de entrega e restrições orçamentárias.
- 86. (CESPE / IPHAN – 2018)** Tanto a etnografia quanto o protótipo podem ser utilizados para validação e elicitação de requisitos, contudo a aplicação de um elimina a possibilidade de uso do outro no mesmo cenário, pois se tratam de técnicas excludentes.



- 87. (CESPE / IPHAN – 2018)** A validação de requisitos se sobrepõe à análise de requisitos, pois tem a finalidade de encontrar eventuais problemas nos requisitos e validá-los conforme as necessidades dos usuários do sistema.
- 88. (CESPE / IPHAN – 2018)** Situação hipotética: Como forma de obter os requisitos de apoio para desenvolver um sistema a ser implementado em determinado setor de uma organização, um analista propôs que se observasse o trabalho do dia a dia, anotando-se as tarefas realizadas no referido setor. Assertiva: Para o cenário proposto, é ideal a utilização da técnica de caso de uso alinhada à entrevista.
- 89. (CESPE / IPHAN – 2018)** Situação hipotética: Na metodologia de desenvolvimento de software customizada para uma organização, o analista propôs o uso da prototipação na fase de engenharia de requisitos, contudo julgou inviável a utilização da prototipação na fase de projeto de sistemas. Assertiva: Nessa situação, a proposta do analista está incorreta, pois a prototipação tanto pode ser utilizada no processo de engenharia de requisitos, para ajudar na elicitação de requisitos, quanto no projeto de sistema, para apoiar o projeto de interface de usuário.
- 90. (CESPE / PF – 2018)** No desenvolvimento de um sistema de informação, a fase de levantamento de requisitos consiste em compreender o problema, dando aos desenvolvedores e usuários a mesma visão do que deve ser construído para resolvê-lo, e a fase de projeto consiste na realização da descrição computacional, incluindo a arquitetura do sistema, a linguagem de programação utilizada e o sistema gerenciador de banco de dados (SGBD) utilizado.
- 91. (CESPE / MPE-PI – 2018)** A análise de requisitos consiste na área responsável pela identificação das reais necessidades dos clientes de TI. Por meio da análise de requisitos, em conjunto com o cliente, é possível construir uma solução que atenda essas necessidades e desenvolver os requisitos funcionais elencados.
- 92. (CESPE / MPE-PI – 2018)** Situação hipotética: Ao se iniciar a especificação de requisitos de um software para controlar o gasto de folhas impressas de um setor, o analista de requisitos, juntamente com o gestor, definiu um cenário de teste em que, ao se comandar a impressão, a chave do usuário autenticado no sistema que comandar uma impressão acionará o contador de impressões do setor de locação desse usuário. Assertiva: Nessa situação, o teste validará o cenário do requisito definido junto com o gestor.
- 93. (CESPE / SE-DF – 2017)** Para auxiliar na gerência de riscos e prevenir insatisfações das partes interessadas, deve-se dificultar as modificações na especificação dos requisitos.
- 94. (CESPE / SE-DF – 2017)** Um dos objetivos da engenharia de requisitos é integrar tarefas, técnicas, orientações, responsabilidades e papéis em fluxos de trabalho.
- 95. (CESPE / SE-DF – 2017)** É comum que uma especificação de requisitos inclua as interfaces externas do software.



96. (CESPE / TRE-PE – 2017) No contexto da análise de requisitos, confiabilidade e usabilidade são atributos de qualidade classificados como:
- a) requisitos funcionais.
  - b) requisitos de domínio.
  - c) requisitos não funcionais.
  - d) dependências.
  - e) regras de negócio.
97. (CESPE / TCE-PR – 2016) Com relação aos requisitos de software, assinale a opção correta.
- a) O documento de especificação de requisitos é um documento restrito à equipe de desenvolvimento de software.
  - b) As necessidades do usuário são informações que substituem os requisitos do software.
  - c) Os requisitos de produto e os requisitos organizacionais são tipos de requisitos funcionais.
  - d) Os requisitos funcionais descrevem as funcionalidades, os recursos e as características do software.
  - e) Os requisitos não funcionais referem-se diretamente às características do software.
98. (CESPE / TRT-PR – 2016 – Letra D) Durante a fase de levantamento de requisitos para a construção de um software, compete aos desenvolvedores organizar as necessidades em ordem de prioridade.
99. (CESPE / TRT-PR – 2016 – Letra E) O QFD (quality function deployment) identifica como requisitos fascinantes os recursos que extrapolam as expectativas dos clientes.
100. (CESPE / MPOG-ATI – 2015) Tão logo exista uma versão do documento de requisitos, o processo de gerenciamento de requisitos deverá ser iniciado.
101. (CESPE / MPOG-ATI – 2015) As informações de rastreabilidade de requisitos possibilitam a realização de estimativa do custo de mudanças em requisitos.
102. (CESPE / MPOG-ATI – 2015) As mudanças de requisitos em processos ágeis de desenvolvimento não seguem um processo formal de gerenciamento de requisitos.
103. (CESPE / MPOG-ATI – 2015) Para a elicitação dos requisitos, é indicada à empresa a realização de um workshop de requisitos, em que seja determinado um facilitador, mesmo que sem grande experiência com os processos de gerenciamento de requisitos.



104. (CESPE / MPOG-ATI – 2015) Os requisitos não funcionais a serem especificados estabelecerão restrições que devem ser seguidas por todo o sistema da referida empresa, podendo até mesmo levar à necessidade de definição de requisitos funcionais.
105. (CESPE / MPOG-ATI – 2015) A definição de um protótipo para a validação dos requisitos pode tornar o processo de requisitos mais barato e mais simplificado, já que ele vai corresponder à real forma de uso do sistema a ser construído.
106. (CESPE / MPOG-ATI – 2015) Uma forma de validação dos requisitos é a geração de casos de teste para os requisitos documentados.
107. (CESPE / MPOG-ATI – 2015) No ciclo de vida do software, o congelamento dos requisitos do software garante que este, quando em desenvolvimento, atenda à expectativa do usuário, desde que tudo que tenha sido requisitado seja implementado.
108. (CESPE/ STJ – 2015) Os requisitos ambientais, operacionais e de desenvolvimento são organizacionais e não funcionais.
109. (CESPE / STJ – 2015) Os requisitos reguladores, legais e éticos são externos e não funcionais.
110. (CESPE / TJDFT – 2015) O uso de protótipo auxilia a descoberta e a validação dos requisitos de software.
111. (CESPE / TJDFT – 2015) As técnicas de elicitação e especificação de requisitos incluem a etnografia, a qual é utilizada para compreender os requisitos sociais e organizacionais para determinado projeto.
112. (CESPE / TJDFT – 2015) É caracterizada como requisito funcional a exigência de que, em determinado projeto, o software desenvolvido funcione no sistema operacional Linux, uma vez que essa exigência está diretamente ligada ao software.
113. (CESPE / MPU – 2013) As atividades do gerenciamento de requisitos incluem a análise e a negociação, a qual visa garantir que todos os requisitos do sistema tenham sido declarados de modo não ambíguo, sem inconsistências, omissões e erros.
114. (CESPE / MPE-PI – 2012) Identificada facilidade do cliente em entender uma especificação matemática, é correto utilizar, também, na especificação dos requisitos, notações baseadas em máquinas de estado finito, uma vez que elas podem reduzir a ambiguidade de um documento de requisitos.
115. (CESPE / EBC – 2011) No processo de construção e (ou) manutenção de um produto de software, o termo requisito pode ser definido da seguinte forma: uma condição, característica





ou capacidade, determinada no universo das necessidades do negócio do usuário, que deve ser atendida por um software na forma de aspectos funcionais e não funcionais.

116. **CESPE / EBC – 2011)** O principal artefato elaborado no processo de produção de requisitos do sistema, segundo a ER, é o documento de requisitos. Por sua vez, o documento de requisitos é uma declaração formal dos requisitos para os stakeholders, que podem ser clientes, usuários finais ou a equipe de desenvolvimento do software.
117. **(CESPE / BRB – 2011)** O levantamento de requisitos de software privilegia a visão do desenvolvedor em relação aos requisitos de um produto. Já a análise dos requisitos prioriza a visão que o cliente e os usuários têm dos requisitos de um produto.
118. **(CESPE / FUB – 2011)** A etnografia, uma técnica de levantamento de requisitos, é uma abordagem completa para elicitacão, utilizada para compreender os requisitos sociais e organizacionais e que identifica novas características a serem acrescentadas em um sistema.
119. **(CESPE / EBC – 2011)** Uma das principais técnicas de verificacão é a prototipacão. Um protótipo é um produto parcialmente desenvolvido, que possibilita aos clientes e desenvolvedores examinarem certos aspectos do sistema proposto e decidir se eles são ou não apropriados ou adequados para o produto acabado.
120. **(CESPE / TJ-ES – 2011)** Assim como o software, os requisitos também devem ser avaliados quanto à qualidade. A validacão, atividade da engenharia de requisitos, é responsável por garantir que os requisitos tenham sido declarados de forma clara e precisa. Além disso, a validacão busca detectar inconsistências, erros e omissões, objetivando alinhar os requisitos às normas estabelecidas para o projeto, produto e processo.
121. **(CESPE / STM – 2011)** São consideradas técnicas de validacão de requisitos: revisões de requisitos, prototipacão e geracão de casos de teste.
122. **(CESPE / TJ-ES – 2011)** Verificacão e validacão são atividades da análise de software, necessárias para se identificar o que o software precisa executar, seguida de uma avaliacaão do usuário quanto às atividades definidas.
123. **(CESPE / MEC – 2011)** A rastreabilidade de requisitos ocorre apenas na relacão entre os requisitos propriamente ditos e os artefatos ou subprodutos de desenvolvimento gerados.
124. **CESPE / ABIN – 2010)** Requisitos não funcionais são restriçoes sobre os serviços ou as funções oferecidas pelo sistema, e podem ser, também, declaraçoes de serviços que o sistema deve fornecer, como o sistema deve reagir a entradas específicas e como deve comportar-se em diversas situaçoes.





- 125. (CESPE / MPU – 2010)** Os requisitos normativos, geralmente oriundos da análise das regras de negócio a que está submetido um sistema, nunca podem ser considerados requisitos funcionais, por estarem fora do sistema, ou seja, do domínio do negócio.
- 126. (CESPE / ABIN – 2010)** Se os requisitos forem organizados de acordo com os diversos pontos de vista relativos a grupos de usuários do sistema, é possível identificar aqueles comuns a todos ou à maioria dos pontos de vista. Esses requisitos comuns podem estar relacionados a assuntos separados, implementados como extensões da funcionalidade central.
- 127. (CESPE / MPU – 2010)** O levantamento de requisitos é realizado ao final da primeira versão de um protótipo, para se definir, junto aos envolvidos no processo, quais são as premissas básicas para o início do entendimento das funcionalidades desejadas.
- 128. (CESPE / MPU – 2010)** Embora a criação de uma sequência ilustrada de telas por meio de programas de desenho gráfico seja útil para a identificação de alguns requisitos do software, ela não é considerada uma atividade de prototipação por não envolver o uso de uma linguagem de programação.
- 129. (CESPE / DETRAN-ES – 2010)** A técnica de brainstorm é adequada para a produção de especificações de requisitos para um sistema de informação em desenvolvimento.
- 130. (CESPE / MPU – 2010)** A verificação de requisitos tem por objetivo analisar se os modelos construídos estão de acordo com os requisitos definidos. Por sua vez, a validação de requisitos visa assegurar que as necessidades do cliente estão sendo atendidas por tais requisitos.
- 131. (CESPE / TCU – 2010)** Por se tratar de função essencial da engenharia de requisitos, a gestão formal de requisitos é indispensável mesmo para projetos de pequeno porte, com apenas duas ou três dezenas de requisitos identificáveis.
- 132. (CESPE / DETRAN-DF – 2009)** Requisitos funcionais são restrições sobre as funções ou serviços oferecidos pelo sistema. Esses requisitos consideram as declarações de serviços, a forma do sistema reagir e como ele deve se comportar em determinadas situações. Cenários e casos de uso são técnicas eficazes para elicitação de requisitos funcionais segundo pontos de vista de interação.
- 133. (CESPE / IPEA – 2009)** Elicitação envolve a identificação sistemática de requisitos nem sempre explicitados pelos clientes. Protótipos, pesquisas estruturadas, testes-beta, análise de casos de negócio, walkthroughs, QFD, grupos de trabalho são exemplos de técnicas utilizadas para elicitar necessidades, expectativas, restrições e interfaces dos stakeholders para todas as fases do ciclo de vida do produto.
- 134. (CESPE / TCE-RN – 2009)** A etnografia é uma técnica utilizada para a descoberta de requisitos de sistemas de software na qual, por meio de observações, procura-se compreender os requisitos sociais e organizacionais do ambiente onde o sistema será usado.



- 135. (CESPE / IPEA – 2009)** A política organizacional para o planejamento e execução do processo de gerenciamento de requisitos reflete as expectativas organizacionais para processos de gestão de requisitos e para que seja possível identificar inconsistências entre os requisitos e os planos do projeto.
- 136. (CESPE / STJ – 2008)** Os requisitos de um sistema podem ser descrições dos serviços fornecidos ou restrições operacionais. Requisitos podem ainda ser classificados como funcionais, não funcionais, ou de domínio. A engenharia de requisitos visa compreender e definir os requisitos. Um processo de engenharia de requisitos pode envolver o estudo de viabilidade, a análise, a especificação e a validação de requisitos.
- 137. (CESPE / SERPRO – 2008)** O levantamento de requisitos é importante, porém não é fundamental, pois recomenda-se avançar o quanto antes para as demais atividades que permitam uma visualização do software e reduzam a ansiedade do cliente em ver algo pronto.
- 138. CESPE / MPE-RR – 2008)** Os requisitos de um sistema são descrições dos serviços fornecidos pelo sistema e suas restrições operacionais. O processo de descobrir, analisar, documentar e verificar esses serviços e restrições é denominado engenharia de requisitos. Requisitos de um sistema de software podem ser funcionais, não funcionais ou de domínio.
- 139. (CESPE / MPU – 2007)** A especificação de requisitos permite, em determinado momento, revelar o que o sistema irá realizar no que se refere às funcionalidades, sem definir, nesse momento, como as funcionalidades serão implementadas.
- 140. (CESPE / MPU – 2007)** Na validação de requisitos — parte integrante da especificação desses requisitos —, é correto o uso de diagramas da UML, tais como diagrama de classes, de casos de uso e de interação.
- 141. (CESPE / SERPRO – 2005)** O gerenciamento de requisitos inclui, entre outras, as seguintes atividades: levantar, analisar, especificar, validar e prototipar requisitos funcionais e não-funcionais.
- 142. (CESPE / SERPRO – 2005)** Uma das principais atividades relacionadas à engenharia de software é o levantamento dos requisitos. Nesse contexto, foi introduzida, na década de 80 do século XX, uma técnica de entrevista conhecida como JAD (Joint Application Development), que consistia em uma rápida entrevista e um processo acelerado de coleta de dados em que todos os principais usuários e o pessoal da análise de sistemas agrupavam-se em uma única e intensiva reunião.
- 143. (CESPE / AGE-ES – 2004)** A engenharia de requisitos fornece mecanismos que permitem entender e analisar a necessidade de o cliente avaliar a exequibilidade, negociar uma solução razoável e especificá-la de maneira não-ambígua, validar a especificação e administrar os requisitos.



- 144. (CESPE / Prefeitura de Boa Vista – 2004)** Requisitos adequadamente definidos constituem base importante sobre a qual um sistema poderá ser bem desenvolvido. No processo de engenharia de requisitos, o estudo de viabilidade utiliza as informações do processo de levantamento de requisitos para gerar um relatório que recomenda se é viável ou não realizar o processo de desenvolvimento do sistema.
- 145. (CESPE / COHAB – 2004)** O QFD (Quality Function Deployment) tem uma abordagem embasada na criação de uma equipe formada por clientes e desenvolvedores, que trabalham juntos para identificar o problema, propor elementos da solução, negociar diferentes abordagens e especificar um conjunto de requisitos da solução.
- 146. (CESPE / COHAB – 2004)** As atividades de análise de requisitos resultam na especificação das características operacionais do software, na indicação da interface do software com outros elementos do sistema e no estabelecimento de restrições que o software deve satisfazer.
- 147. (CESPE / COHAB – 2004)** À medida que os requisitos são elucidados, o analista de software pode criar um conjunto de cenários, ou seja, casos de uso, que identificam uma linha de uso para o sistema a ser construído.



## GABARITO

- |     |         |     |         |      |         |
|-----|---------|-----|---------|------|---------|
| 1.  | ERRADO  | 41. | ERRADO  | 81.  | CORRETO |
| 2.  | CORRETO | 42. | CORRETO | 82.  | ERRADO  |
| 3.  | ERRADO  | 43. | ANULADO | 83.  | ERRADO  |
| 4.  | LETRA B | 44. | CORRETO | 84.  | CORRETO |
| 5.  | CORRETO | 45. | ERRADO  | 85.  | ERRADO  |
| 6.  | ANULADO | 46. | CORRETO | 86.  | ERRADO  |
| 7.  | CORRETO | 47. | CORRETO | 87.  | CORRETO |
| 8.  | CORRETO | 48. | ERRADO  | 88.  | ERRADO  |
| 9.  | CORRETO | 49. | CORRETO | 89.  | CORRETO |
| 10. | ANULADO | 50. | CORRETO | 90.  | CORRETO |
| 11. | CORRETO | 51. | ERRADO  | 91.  | CORRETO |
| 12. | LETRA D | 52. | CORRETO | 92.  | CORRETO |
| 13. | CORRETO | 53. | CORRETO | 93.  | LETRA E |
| 14. | ERRADO  | 54. | ERRADO  | 94.  | CORRETO |
| 15. | LETRA C | 55. | ERRADO  | 95.  | CORRETO |
| 16. | ERRADO  | 56. | ERRADO  | 96.  | LETRA C |
| 17. | CORRETO | 57. | CORRETO | 97.  | LETRA D |
| 18. | ERRADO  | 58. | ERRADO  | 98.  | ERRADO  |
| 19. | CORRETO | 59. | CORRETO | 99.  | CORRETO |
| 20. | CORRETO | 60. | CORRETO | 100. | CORRETO |
| 21. | ERRADO  | 61. | ERRADO  | 101. | CORRETO |
| 22. | CORRETO | 62. | LETRA D | 102. | CORRETO |
| 23. | ERRADO  | 63. | LETRA A | 103. | CORRETO |
| 24. | CORRETO | 64. | CORRETO | 104. | CORRETO |
| 25. | CORRETO | 65. | ERRADO  | 105. | ERRADO  |
| 26. | ERRADO  | 66. | CORRETO | 106. | CORRETO |
| 27. | LETRA A | 67. | CORRETO | 107. | ERRADO  |
| 28. | CORRETO | 68. | ERRADO  | 108. | CORRETO |
| 29. | CORRETO | 69. | ERRADO  | 109. | CORRETO |
| 30. | ERRADO  | 70. | CORRETO | 110. | CORRETO |
| 31. | CORRETO | 71. | ERRADO  | 111. | CORRETO |
| 32. | ERRADO  | 72. | ERRADO  | 112. | ERRADO  |
| 33. | ERRADO  | 73. | CORRETO | 113. | LETRA E |
| 34. | CORRETO | 74. | CORRETO | 114. | CORRETO |
| 35. | ERRADO  | 75. | ERRADO  | 115. | CORRETO |
| 36. | CORRETO | 76. | ERRADO  | 116. | CORRETO |
| 37. | CORRETO | 77. | ERRADO  | 117. | ERRADO  |
| 38. | CORRETO | 78. | CORRETO | 118. | ERRADO  |
| 39. | CORRETO | 79. | ERRADO  | 119. | ERRADO  |
| 40. | ERRADO  | 80. | ERRADO  | 120. | CORRETO |



- 121. CORRETO
- 122. CORRETO
- 123. ERRADO
- 124. ERRADO
- 125. ERRADO
- 126. ERRADO
- 127. ERRADO
- 128. ERRADO
- 129. ERRADO
- 130. CORRETO
- 131. ERRADO
- 132. ERRADO
- 133. CORRETO
- 134. CORRETO
- 135. CORRETO
- 136. CORRETO
- 137. ERRADO
- 138. CORRETO
- 139. CORRETO
- 140. CORRETO
- 141. CORRETO
- 142. CORRETO
- 143. CORRETO
- 144. ERRADO
- 145. ERRADO
- 146. CORRETO
- 147. CORRETO





## LISTA DE QUESTÕES – FCC

1. (FCC / SEFAZ-AP – 2022) Considere as seguintes especificações de requisitos de software:
  - I. O sistema deve calcular a dívida do contribuinte aplicando a alíquota de 15% quando o lucro ultrapassar o teto de contribuição.
  - II. O tempo de resposta da consulta à dívida ativa da empresa não deve ultrapassar os 13 ms em situações normais de processamento.
  - III. O SLA (Acordo de Nível de Serviço) com o contribuinte consulente deve prever jornada de 24 horas/dia × 7 dias por semana.
  - IV. A tela de consulta à dívida ativa só pode ser acessada mediante login e senha corretos correspondentes àqueles designados ao CNPJ do contribuinte consulente.Esses requisitos são, correta e respectivamente, dos tipos
  - a) funcional, técnico, de usuário e não funcional.
  - b) funcional, não funcional, não funcional e funcional.
  - c) funcional, técnico, de sistema e não funcional.
  - d) não funcional, não funcional, técnico e de sistema.
  - e) não funcional, de usuário, técnico e funcional.
2. (FCC / AL-AP – 2020) Considere a lista abaixo, elaborada durante um levantamento de requisitos na Assembleia Legislativa do Amapá, para um sistema hipotético de avaliações internas:
  1. Registrar avaliação de colaborador por parlamentar: O sistema deve permitir ao parlamentar, em uma única tela, a avaliação de todos os seus colaboradores.
  2. Considerar Aspectos Legais: O sistema deve seguir orientações elencadas na Resolução 099/XXXX do Conselho Legislativo do Estado.
  3. Registrar autoavaliação de parlamentar: O sistema deve permitir ao parlamentar sua autoavaliação em relação às disposições legais sob as quais atuou no período.
  4. Atentar à Segurança: O sistema deve fornecer mecanismos de segurança e autenticação alinhados com os adotados pelo processo XPTO.
  5. Impedir acesso direto ao processo XPTO: O sistema deverá mostrar ao usuário que existem formulários de avaliação a serem respondidos e dará a opção de respondê-los depois.





Adotando RFU para requisitos funcionais e RNF para não-funcionais, a classificação correta e respectiva da lista 1 a 5 acima é:

- a) RFU, RFU, RFU, RNF e RNF.
- b) RFU, RNF, RFU, RNF e RNF.
- c) RFU, RNF, RFU, RNF e RFU.
- d) RNF, RNF, RFU, RNF e RFU.
- e) RNF, RFU, RFU, RNF e RNF.

3. (FCC / TRT-19 – 2019) A Engenharia de Requisitos utiliza algumas técnicas que apoiam as atividades de levantamento de requisitos, sendo a entrevista uma das mais utilizadas. Uma entrevista pode ser estruturada de formas diferentes, como na estrutura em:

a) diamante, que envolve sessões de workshop com os usuários os quais assumem papéis de documentadores, escrevendo os requisitos em flipcharts.

b) brainstorming, em que inicia-se com perguntas mais genéricas sobre o sistema e finaliza-se com perguntas mais específicas, sendo geralmente utilizada com usuários que desconhecem o assunto.

c) funil, na qual procura-se manter o usuário interessado no assunto e para isto utilizam-se perguntas variadas sobre o sistema, sorteadas com um dado.

d) diamante, na qual os usuários escrevem os requisitos em papel, todos ao mesmo tempo, em uma tempestade de ideias, para estimular requisitos criativos.

e) pirâmide, em que inicia-se com perguntas mais específicas sobre o sistema e finaliza-se com perguntas mais genéricas, sendo geralmente utilizada com usuários mais relutantes.

4. (FCC / AFAP – 2019) Um Analista de Informática levantou os requisitos para desenvolver um sistema de gestão. Dentre os requisitos levantados,

I. o sistema deve apresentar a tela de login e senha antes de cada transação e validar o acesso com base nas políticas de segurança organizacional.

II. o sistema deve estar disponível para a diretoria em tempo integral, ou seja, 24 x 7.

III. o tempo de resposta de uma consulta da alta administração não pode exceder a 5 milissegundos.

IV. cada Diretor que usa o sistema deve ser identificado apenas por sua matrícula de cinco dígitos seguidos do código de segurança.



V. o sistema deverá gravar um log de autenticação a cada transação completada, contendo a identificação do usuário, data e equipamento utilizado.

VI. os backups do sistema deverão ser feitos diariamente a fim de evitar a eventual perda de dados sem capacidade de recuperação.

Contêm um requisito funcional e um requisito não funcional, respectivamente, APENAS os itens

- a) II e I.
- b) V e I.
- c) IV e VI.
- d) II e III.
- e) VI e V.

5. (FCC / TRF4 – 2019) Suponha que um Analista de TI, participando da etapa de análise de requisitos de um sistema de emissão de certidão negativa para o TRF4, tenha elencado os requisitos apresentados abaixo:

1. Utilizar interface responsiva para que possa ser executado em dispositivos móveis e na web.
2. Validar o tipo de certidão solicitado.
3. Emitir certidão negativa após verificação de situação do requerente.
4. Solicitar o CPF do requerente.
5. Responder ao clique único do usuário em qualquer botão da interface.
6. Validar o CPF do requerente.
7. Restaurar os dados automaticamente após falhas não programadas.
8. Solicitar o nome do requerente.
9. Oferecer dois tipos de certidão: para fins gerais e para fins eleitorais.
10. Emitir aviso de impossibilidade de emissão da certidão.

Sobre os requisitos, é correto afirmar que:

- a) todos são funcionais.
- b) todos são não funcionais.
- c) 1, 5 e 7 são não funcionais.
- d) apenas 3, 4, 8, 9 e 10 são funcionais.
- e) apenas 2, 6 e 7 são não funcionais.

6. (FCC / SAMASA Campinas – 2019) O diagrama faz referência à QFD –





a) Quality Function Deployment, uma técnica da gestão de qualidade que traduz as necessidades do cliente para requisitos de software, buscando maximizar a sua satisfação.

b) Questionário de Funcionalidades para Desenvolvimento, uma técnica para priorização de requisitos que facilita a criação de casos de uso.

c) Questionnaire For Diagram, uma lista de perguntas que ajudam a entender melhor o problema e permitem que o cliente expresse os requisitos essenciais para a criação de diagramas de caso de uso.

d) Quality Function Development, uma técnica para priorização de requisitos, especializada para a criação de casos de uso.

e) Questionário de Funcionalidades para Desenvolvimento, uma técnica da gestão de qualidade que traduz as necessidades do cliente para as funcionalidades a serem incorporadas no software.

7. (FCC / SEFAZ-BA – 2019) Um profissional da área administrativa de certa instituição recebeu um Analista de Sistemas que estava fazendo o levantamento de requisitos para a construção de um novo software. Ao informar ao Analista um requisito não funcional para seu departamento, o profissional corretamente disse que:

a) a resposta a uma consulta de dados deveria durar no máximo dois segundos para não atrasar seu trabalho.



- b) o sistema deveria permitir a alteração de dados incluídos de forma equivocada.
- c) o acesso ao sistema deveria ser por meio de uma senha composta por letras e números e não apenas por números.
- d) o sistema deveria permitir a exclusão de registros de pessoas que deixaram de ser clientes da instituição.
- e) o sistema, após consultar os dados de um cliente, deveria permitir a impressão dos dados.

8. (FCC / SEFAZ-BA – 2019) Um Auditor Fiscal da área de Tecnologia da Informação está participando do processo de levantamento de requisitos para o desenvolvimento de um novo software. Os requisitos a seguir foram elencados:

- I. Um usuário deve ser capaz de pesquisar a lista de contribuintes devedores.
- II. O sistema deve gerar a lista de contribuintes com atendimento agendado naquele dia.
- III. O sistema deve se adequar às leis que garantem o sigilo das informações.
- IV. Cada usuário do sistema deverá ser identificado por um número de 8 dígitos.
- V. O Sistema deve ter suporte para os sistemas operacionais Linux e Windows.
- VI. A alteração dos dados de um contribuinte só poderá ser concretizada após confirmação.
- VII. Toda consulta deve retornar os valores solicitados em até 20 segundos.
- VIII. A gravação dos dados só deverá ser efetuada após o preenchimento de todos os campos de preenchimento obrigatório.
- IX. Os dados devem ser armazenados em servidores em cluster para garantir a disponibilidade.

São requisitos funcionais os que constam APENAS em:

- a) I, II, III, IV, V, VI e VII.
- b) II, IV, VII e VIII.
- c) I, II, IV, VI e VIII.
- d) IV, VI, VIII e IX.
- e) II, IV, V, VI e VIII.

9. (FCC / SEMEF-AM – 2019) Considerando que a Fazenda Municipal emprega o gerenciamento de requisitos, ganha importância o cuidado com os chamados requisitos voláteis, dentre os



quais fazem parte os requisitos que surgem à medida que o cliente vai aprimorando sua compreensão do sistema, denominados requisitos:

- a) mutantes.
- b) de compatibilidade
- c) emergentes
- d) adaptativos
- e) secundários

**10. (FCC / SEMEF-AM – 2019)** Ao fazer uso da engenharia de requisitos em projetos, deve-se analisar o processo de elicitação e análise de requisitos, o qual pode ser dividido nas seguintes atividades:

- I. Documentação de Requisitos.
- II. Classificação e Organização de Requisitos.
- III. Obtenção de Requisitos.
- IV. Priorização e Negociação de Requisitos.

A ordem sequencial correta para a execução dessas atividades é:

- a) I, III, IV e II.
- b) II, IV, III e I.
- c) III, II, IV e I.
- d) IV, I, II e III.
- e) III, I, II e IV.

**11. (FCC / SEMEF-AM – 2019)** O processo de validação de requisitos de software deve ser utilizado em um projeto da Fazenda Municipal, sendo que seus técnicos de TI, devem, nesse processo de validação, efetuar revisões de requisitos, atentando que a propriedade:

- a) facilidade de compreensão analisa se o requisito pode ser excluído sem prejuízo ao sistema.
- b) adaptabilidade verifica se o requisito pode ser alterado sem afetar, de forma significativa, os demais requisitos.
- c) rastreabilidade verifica se o requisito pode ser testado, de forma completa.
- d) facilidade de verificação examina se requisito pode ser excluído sem prejuízo ao sistema.
- e) facilidade de compreensão analisa se o requisito tem sua origem diretamente estabelecida.

**12. (FCC / Prefeitura de Manaus-AM – 2019)** Considerando a análise de requisitos, as informações de rastreabilidade desempenham papel de grande importância. Assim, a equipe responsável da Fazenda Municipal deve estar ciente de que a rastreabilidade de projeto significa:

- a) definir o mapeamento entre os requisitos de projeto e os usuários do sistema.
- b) listar os compiladores utilizados no desenvolvimento de cada módulo de software.



- c) determinar o mapeamento entre os requisitos de projeto e os locais onde o sistema será utilizado.
- d) determinar o desempenho de cada um dos requisitos do sistema.
- e) possuir o mapeamento entre os requisitos e os módulos de projeto que implementam os requisitos.

**13. (FCC / SEFAZ-SC – 2018)** A definição de contextos para que os usuários possam agir de maneira semelhante, entendendo melhor quais informações precisam fornecer durante a atividade de elicitação de requisitos, pode ser obtida por meio da aplicação de duas técnicas de elicitação denominadas:

- a) cenários e protótipos.
- b) entrevistas e observação.
- c) protótipos e observação.
- d) cenários e histórias de usuários.
- e) reuniões com facilitadores e histórias de usuários.

**14. (FCC / SEFAZ-SC – 2018)** Durante o processo de validação, diferentes tipos de verificação podem ser efetuados com os requisitos registrados nos documentos de requisitos. O tipo de verificações de consistência é realizado para:

- a) identificar, por meio de análise mais aprofundada, outras funções necessárias, adicionais ou diferentes, além daquelas que um usuário pensava que fossem as necessárias para o sistema executar determinadas funções.
- b) evitar que requisitos, no documento, entrem em conflito uns com outros, ou seja, não deve haver restrições contraditórias ou descrições diferentes para mesma função do sistema.
- c) garantir que o documento de requisitos contenha os requisitos que definem todas as funções e as restrições pretendidas pelos usuários do sistema.
- d) assegurar, usando o conhecimento das tecnologias existentes, que os requisitos verificados possam ser realmente implementados, considerando o orçamento e o cronograma para o desenvolvimento do sistema.
- e) reduzir o potencial de conflito entre o cliente e o contratante por meio de um conjunto de testes que demonstre que o sistema entregue atende a cada requisito especificado.

**15. (FCC / SABESP – 2018)** Um Analista necessita levantar os requisitos de um sistema junto aos usuários. São técnicas de levantamento:

- a) Cenários e Peer Review.
- b) Product Owner e Brainstorming.
- c) Overview e Use Cases.



- d) Joint Application Design (ou Development) – JAD e Etnografia.
- e) Prototipação e Sprint.

**16.(FCC / FUB – 2018)** O documento de requisitos deve ser elaborado a partir da análise de viabilidade do software, seguida de análise, especificação e validação de requisitos.

**17.(FCC / BNB – 2018)** A revisão técnica é um procedimento utilizado para validar os requisitos de um projeto, com o objetivo de identificar eventuais inconsistências e verificar se os artefatos estão de acordo com o padrão esperado.

**18.(FCC / DPE-AM – 2018)** Considere, por hipótese, que uma equipe de Analistas de Sistemas da Defensoria elencou a lista de requisitos para um novo sistema:

- O sistema não deverá revelar aos usuários nenhuma informação pessoal sobre os cidadãos, além do número do processo, em respeito à legislação de privacidade.
- Em razão das restrições referentes aos direitos autorais, alguns documentos devem ser excluídos imediatamente ao serem fornecidos pelos cidadãos em seus processos.
- O sistema deve implementar interfaces utilizando as normas de usabilidade vigentes para o serviço público.

A lista apresenta exemplos de requisitos:

- a) funcionais do tipo proteção e do tipo regulação.
- b) funcionais de usabilidade.
- c) não-funcionais de proteção.
- d) funcionais internos de legislação.
- e) não-funcionais externos do tipo legal e do tipo regulador.

**19.(FCC / BNB – 2018)** No levantamento de informações, os requisitos dos solicitantes são classificados como normais e conceituais. Os requisitos normais refletem os objetivos e as metas do produto, ao passo que os conceituais estão implícitos no produto ou extrapolam as expectativas do cliente.

**20.(FCC / BNB – 2018)** O protótipo operacional serve para aprimorar o entendimento de como o sistema deve funcionar, por meio da elucidação dos requisitos do usuário e da compreensão de suas necessidades.

**21.(FCC / MPE-MA – 2013)** O escopo de um projeto é determinado pelo levantamento de requisitos funcionais e não funcionais. Dentre os requisitos não funcionais se enquadram os requisitos organizacionais, que podem ser divididos em:

- a) reguladores e éticos.





- b) ambientais, operacionais e de desenvolvimento.
- c) contábeis e de segurança.
- d) de desempenho e de espaço.
- e) de eficiência, de confiança e de proteção.

**22. (FCC / DPE-SP – 2013)** Em uma das etapas da Engenharia de Requisitos há a preocupação em se observar a especificação produzida, visando verificar que os requisitos tenham sido declarados, por exemplo, sem ambiguidades.

O texto refere-se à etapa de:

- a) gestão dos requisitos.
- b) elicitação dos requisitos.
- c) negociação dos requisitos.
- d) levantamento dos requisitos.
- e) validação dos requisitos.

**23. (FCC / TST – 2012)** Na Engenharia de Requisitos, o gerente de requisitos:

- a) acompanha e monitora ações durante a verificação do software, sendo este o processo que garante o atendimento aos requisitos informados pelo usuário final.
- b) possui autonomia para realizar alterações no projeto para garantir que o software seja bem construído e atenda às necessidades da equipe de desenvolvimento.
- c) mantém atualizados os requisitos junto ao usuário final e a equipe de desenvolvimento, a fim de obter sucesso no processo de homologação do software, atendendo as necessidades e expectativas.
- d) classifica os requisitos em diferentes tipos, sendo os do tipo funcional relacionados com o custo e confiabilidade do software e os do tipo não-funcional relacionados com os casos de uso.
- e) obtém o comprometimento dos integrantes da equipe de desenvolvimento de software para o cumprimento do processo de software.

**24. (FCC / TCE-AP – 2012)** Em relação a requisitos de sistemas, considere:

- I. O modo como um sistema deve reagir a certas entradas e o comportamento em que o sistema deve ter em certas situações e, em alguns casos, especificar o que o sistema não deve fazer, são chamados de requisitos não-funcionais.
- II. As restrições aos serviços ou funções de um sistema, como, por exemplo, processos de desenvolvimento ou utilização de padrões, são requisitos de funcionamento do sistema ou requisitos funcionais.



III. Requisitos que vem do domínio da aplicação do sistema e refletem características ou restrições para aquele domínio são chamados de requisitos de domínio e podem ser requisitos funcionais e/ou não-funcionais.

Está correto o que se afirma em:

- a) III, apenas.
- b) I, II e III.
- c) I e II, apenas.
- d) II e III, apenas.
- e) I, apenas.

**25. (FCC / MPE-PE – 2012)** Os requisitos não funcionais não estão diretamente ligados aos serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área, entre outros. Dentre os tipos de requisitos não funcionais, é possível destacar os requisitos de produto, organizacionais e externos. Dentre os requisitos de produto, podemos citar os requisitos:

- a) de eficiência e de confiança.
- b) contábeis e de desempenho.
- c) legais e de usabilidade.
- d) reguladores e de proteção.
- e) legais e contábeis.

**26. (FCC / TRE-CE – 2012)** Considere:

- I. Para cada cliente deve ser aplicado um identificador único.
- II. O tempo de resposta entre a requisição e a informação não pode exceder a 2 ms.
- III. Clientes têm filiais que devem "carregar", na base de dados, o identificador do cliente principal.
- IV. O sistema não deve ferir as leis de proteção ambiental.

São requisitos não funcionais os que constam em

- a) I e II, apenas.
- b) II e III, apenas.
- c) II e IV, apenas.
- d) I, III e IV, apenas.
- e) I, II, III e IV.



**27. (FCC / TST – 2012)** Na Engenharia de Requisitos, o gerente de requisitos:

- a) acompanha e monitora ações durante a verificação do software, sendo este o processo que garante o atendimento aos requisitos informados pelo usuário final.
- b) possui autonomia para realizar alterações no projeto para garantir que o software seja bem construído e atenda às necessidades da equipe de desenvolvimento.
- c) mantém atualizados os requisitos junto ao usuário final e a equipe de desenvolvimento, a fim de obter sucesso no processo de homologação do software, atendendo as necessidades e expectativas.
- d) classifica os requisitos em diferentes tipos, sendo os do tipo funcional relacionados com o custo e confiabilidade do software e os do tipo não-funcional relacionados com os casos de uso.
- e) obtém o comprometimento dos integrantes da equipe de desenvolvimento de software para o cumprimento do processo de software.

**28. (FCC / TJ-PE – 2012)** Na engenharia de requisitos trata-se de uma técnica de elicitação que ocorre em ambiente mais informal em que toda a idéia deve ser levada em consideração para a solução de um problema, sendo proibida a crítica a qualquer sugestão dada, e encorajada, inclusive, a criação de ideias que pareçam estranhas ou exóticas:

- a) Prototipação.
- b) Entrevista.
- c) Questionário.
- d) Brainstorming.
- e) Análise de protocolos.

**29. (FCC / INFRAERO – 2011 – Letra D)** No contexto de levantamento de requisitos, funcionalidade é um dos aspectos que deve ser levado em conta na abordagem dos requisitos funcionais.

**30. (FCC / INFRAERO – 2011)** A engenharia de requisitos ajuda os engenheiros de software a compreender melhor o problema que eles vão trabalhar para resolver. Ela inclui um conjunto de tarefas que levam a um entendimento de qual será o impacto do software sobre o negócio, do que o cliente quer e de como os usuários finais vão interagir com o software. A função de negociação no processo de engenharia de requisitos:

- a) especifica, revisa e valida o problema de modo a garantir que seu entendimento e o entendimento do cliente sobre o problema coincidam.
- b) refina e modifica os requisitos. É uma ação de modelagem de análise composta de várias tarefas de modelagem e refinamento.



c) define quais são as prioridades, o que é essencial, o que é necessário. Clientes, usuários e outros interessados são solicitados a ordenar os requisitos e depois discutir os conflitos de prioridade.

d) ajuda o cliente a definir o que é necessário.

e) define o escopo e a natureza do problema a ser resolvido.

**31. (FCC / TRT1 – 2011)** A técnica utilizada na compreensão de requisitos sociais e organizacionais por observação das rotinas dos envolvidos é a:

a) prototipação.

b) por pontos de vista.

c) por cenário.

d) entrevista.

e) etnografia.

**32. (FCC / INFRAERO – 2011)** Os produtos de trabalho resultantes da engenharia de requisitos são avaliados quanto à qualidade durante a etapa de validação de requisitos. Analise os itens a seguir referentes a essa etapa:

I. Um dos principais mecanismos de validação de requisitos é a avaliação técnica formal.

II. O modelo de análise pode garantir que os requisitos foram consistentemente declarados.

III. É frequentemente útil examinar cada requisito em face de um conjunto de questões do tipo checklist.

IV. A equipe de revisão que avalia os requisitos inclui apenas pessoas com conhecimento técnico na área de TI, como engenheiros de softwares, desenvolvedores etc.

Está correto o que consta em:

a) I, II, III e IV.

b) II e IV, apenas.

c) I, II e IV, apenas.

d) II, III e IV, apenas.

e) I, II e III, apenas.

**33. (FCC / TRT19 – 2011)** A avaliação do impacto de mudança de um requisito, muitas vezes, faz com que seja necessário retornar à sua fonte. Na validação dos requisitos, a equipe deve estar atenta, portanto, à:



- a) rastreabilidade.
- b) adaptabilidade.
- c) qualidade.
- d) facilidade de compreensão.
- e) facilidade de verificação.

**34. (FCC / TRT23 – 2011)** Tabelas de rastreamento para relacionar os requisitos identificados a um ou mais aspectos do sistema ou do seu ambiente devem ser desenvolvidas, segundo Pressman, na engenharia de requisitos por meio da função de:

- a) gestão.
- b) especificação.
- c) elaboração.
- d) negociação.
- e) validação.

**35. (FCC / BAHIA GÁS – 2010)** É uma restrição sobre os serviços ou as funções oferecidos pelo sistema. Pode ser uma restrição de timing, sobre o processo de desenvolvimento, sobre o desempenho ou sobre a confiabilidade do sistema, entre outras. Trata-se de:

- a) requisito não funcional.
- b) requisito funcional.
- c) especificação de risco.
- d) iteração de processo.
- e) etnografia.

**36. (FCC / DPE-SP – 2010)** Sobre análise de requisitos da engenharia de software, considere:

- I. Os requisitos de usuário podem descrever tanto requisitos funcionais quanto requisitos não-funcionais.
- II. Os requisitos de sistema podem descrever apenas requisitos não funcionais.
- III. Os requisitos não-funcionais podem ser divididos em requisitos de produto, organizacionais e externos.

Está correto o que se afirma em:

- a) III, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.



**37. (FCC / DPE-SP – 2010)** No contexto da Engenharia de Requisitos, considere:

- I. O sistema deve fornecer uma entrada de dados que possibilite a inclusão de atributos de permissão de acesso às dependências da corporação por técnicos, supervisores e chefes.
- II. Algumas permissões de acesso deverão ter tratamento especial para a entrada de atributos. Para este tipo de permissão, atributos excedentes a uma faixa predeterminada só poderão ser incluídos por chefes de seção.

Em relação às assertivas acima, é correto afirmar:

- a) O item I trata de um requisito funcional e a ele está associado o requisito não funcional, contido no item II.
- b) O item I trata de um requisito não funcional e a ele está associado o requisito funcional, contido no item II.
- c) Ambos referem-se a requisitos funcionais.
- d) A assertiva contida no item II é uma condição restritiva do requisito não funcional do item I. Por si só, não constitui um requisito, tanto funcional quanto não funcional.
- e) A assertiva contida no item II é uma condição restritiva do requisito funcional do item I. Por si só, não constitui um requisito, tanto funcional quanto não funcional.

**38. (FCC / MPE-RN – 2010)** Na engenharia de software, etnografia é:

- a) uma fase do processo de software aplicada no modelo em cascata.
- b) uma fase do processo de software aplicada no modelo em espiral.
- c) uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais.
- d) uma técnica aplicada na engenharia de requisitos cujo objetivo é definir, a priori, as classes que contém elementos gráficos (BLOB).
- e) um projeto cujo principal objetivo é criar interfaces gráficas, que facilitam o acesso do usuário (GUI).

**39. (FCC / MPE-RN – 2010)** As políticas de rastreabilidade de requisitos são decididas durante o estágio de:

- a) agregação dos requisitos funcionais, apenas.



- b) implementação do sistema, apenas.
- c) implementação do sistema
- d) eliminação dos requisitos não funcionais.
- e) gerenciamento de requisitos.

**40. (FCC / SEFAZ-SP – 2009)** *É necessário que o software calcule os salários dos diaristas e mensalistas e emita relatórios mensais sumariados por tipo de salário. Entretanto, a base de dados deve estar protegida e com acesso restrito aos usuários autorizados. De qualquer forma, o tempo de resposta das consultas não deve superar os quinze segundos, pois inviabilizaria todo o investimento nesse sistema. Devo lembrar que os relatórios individuais dos departamentos, nos quais constam os salários dos funcionários, devem ser emitidos quinzenalmente em razão dos adiantamentos e vales que recebem. É fundamental que o software seja operacionalizado usando código aberto. Necessito, ainda, forte gerenciamento de risco, prazo e custo, porque a entrega do produto final não pode ultrapassar o prazo de oito meses a contar da data de início do projeto. No texto, são requisitos funcionais:*

- a) Calcule os salários dos diaristas e mensalistas e os relatórios individuais dos departamentos, nos quais constam os salários dos funcionários, devem ser emitidos quinzenalmente.
- b) Necessito, ainda, forte gerenciamento de risco, prazo e custo e a base de dados deve estar protegida e com acesso restrito aos usuários autorizados.
- c) É fundamental que o software seja operacionalizado usando código aberto e emita relatórios mensais sumariados por tipo de salário.
- d) Emita relatórios mensais sumariados por tipo de salário e necessito, ainda, forte gerenciamento de risco, prazo e custo.
- e) A base de dados deve estar protegida e com acesso restrito aos usuários autorizados e entrega do produto final não pode ultrapassar o prazo de oito meses.

**41. (FCC / TRT3 – 2009)** Com relação aos requisitos de software, considere:

- I. funcionais são somente requisitos de usuário.
- II. funcionais e não-funcionais podem ser requisitos de usuário.
- III. funcionais e não-funcionais podem ser requisitos de sistema.

Está correto o que se afirma APENAS em

- a) I.
- b) II.
- c) III.
- d) I e III.
- e) II e III.





**42. (FCC / TRT7 – 2009)** No processo de engenharia de requisitos, é uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais. Trata-se de:

- a) Workshop.
- b) Brainstorming.
- c) Scrum.
- d) Análise de ponto de vista.
- e) Etnografia.

**43. (FCC / TRT3 – 2009)** São técnicas e abordagens utilizadas na obtenção dos requisitos:

- a) estresse, cenários e workshop.
- b) workshop, etnografia e estresse.
- c) etnografia, questionários e validação.
- d) pontos de vista, cenários e entrevista.
- e) pontos de vista, casos de uso e validação.

**44. (FCC / PGE-RJ – 2009)** No âmbito da Engenharia de Requisitos, uma revisão técnica formal é:

- a) um teste de desempenho.
- b) uma técnica de elicitação.
- c) um instrumento de rastreamento.
- d) o resultado do escopo.
- e) um mecanismo de validação.



## GABARITO

- |     |         |     |         |     |         |
|-----|---------|-----|---------|-----|---------|
| 1.  | LETRA B | 16. | CORRETO | 31. | LETRA E |
| 2.  | LETRA C | 17. | CORRETO | 32. | LETRA E |
| 3.  | LETRA E | 18. | LETRA E | 33. | LETRA A |
| 4.  | LETRA C | 19. | ERRADO  | 34. | LETRA A |
| 5.  | LETRA C | 20. | CORRETO | 35. | LETRA A |
| 6.  | LETRA A | 21. | LETRA B | 36. | LETRA C |
| 7.  | LETRA A | 22. | LETRA E | 37. | LETRA A |
| 8.  | LETRA C | 23. | LETRA C | 38. | LETRA C |
| 9.  | LETRA C | 24. | LETRA A | 39. | LETRA E |
| 10. | LETRA C | 25. | LETRA A | 40. | LETRA A |
| 11. | LETRA B | 26. | LETRA C | 41. | LETRA E |
| 12. | LETRA E | 27. | LETRA C | 42. | LETRA E |
| 13. | LETRA A | 28. | LETRA D | 43. | LETRA D |
| 14. | LETRA B | 29. | CORRETO | 44. | LETRA E |
| 15. | LETRA D | 30. | LETRA C |     |         |



## LISTA DE QUESTÕES – FGV

1. (FGV / TJ-AP – 2024) A equipe de desenvolvimento de sistemas do TJAP está implementando uma aplicação de acordo com o Processo Unificado (PU). A equipe atribuiu cada disciplina do PU a um analista diferente. O analista José ficou responsável pela disciplina do PU que é iniciada no começo da fase de concepção, intensificada ao máximo na fase de elaboração e amenizada ao longo da fase de construção. José ficou responsável pela disciplina do PU de:

- a) teste;
- b) análise;
- c) projeto;
- d) requisitos;
- e) modelagem.

2. (FGV / Câmara de Fortaleza – 2024) No contexto de requisitos de software, analise os itens a seguir.

- I. "O sistema deve cadastrar um usuário" é um tipo de Requisito Funcional.
- II. "O sistema deve ficar no ar em 99,9% do tempo" é um tipo de Requisito Funcional de Desempenho.
- III. "O sistema deve ser capaz de armazenar dados de 100 mil clientes" é um tipo de Requisito Não-Funcional de Escalabilidade.

Está correto o que se afirma em:

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

3. (FGV / SES-MT – 2024) Relacione as técnicas de elicitação e especificação de requisitos às suas respectivas definições:

- 1. Entrevistas
- 2. Etnografia
- 3. Histórias e Cenários
- 4. Casos de Uso

( ) Identifica os atores envolvidos, as funcionalidades principais, bem como a interação entre atores e funcionalidades do sistema



( ) Identifica características sobre o sistema (ou processo) atual e sobre o sistema que será desenvolvido a partir de questionamento aos stakeholders

( ) Descreve situação de uso do sistema, com informações como: ator(es); pré-condição; fluxo normal; fluxos alternativos; e pós-condição.

( ) Técnica de observação onde o analista (engenheiro de requisitos) se insere na organização do cliente, observa o trabalho no dia a dia e anota as tarefas dos funcionários

Assinale a opção que indica a relação correta, segundo a ordem apresentada.

a) 2, 4, 1 e 3.

b) 3, 2, 4 e 1.

c) 1, 3, 2 e 4.

d) 4, 1, 3 e 2.

4. (FGV / SES-MT – 2024) Com relação à Engenharia de Requisitos, analise as afirmativas a seguir e assinale (V) para a verdadeira e (F) para a falsa.

( ) A Engenharia de Requisitos inclui três atividades fundamentais: elicitação (e análise); especificação; e, validação.

( ) Na etapa de classificação e organização, no processo de elicitação e análise de requisitos, o conjunto não estruturado de requisitos é agrupado pelos requisitos relacionados e organizado em grupos coerentes.

( ) Na prática, a Engenharia de Requisitos é um processo iterativo, no qual as atividades são intercaladas

As afirmativas são, respectivamente:

a) V, F e F.

b) F, V e F.

c) F, F e V.

d) V, V e V.

5. (FGV / AL-TO – 2024) Sobre Engenharia de Requisitos, analise as afirmativas a seguir.

I. A validação de requisitos é um processo que visa assegurar que os requisitos atendem às necessidades e expectativas dos stakeholders.

II. As técnicas de elicitação de requisitos, como brainstorming, entrevistas e prototipagem, são utilizadas para coletar requisitos de forma passiva, sem interação com os stakeholders.



III. A priorização de requisitos ajuda a definir a ordem de importância para o projeto.

Está correto o que se afirma em:

- a) II, apenas.
- b) I e III, apenas.
- c) II e III, apenas.
- d) I e II, apenas.
- e) I, II e III.

6. (FGV / AL-TO – 2024) A sociedade empresária Max Tech produz um aplicativo de gerenciamento de tarefas para equipes remotas. Após o lançamento, vários usuários relataram problemas de usabilidade e desempenho do aplicativo, especialmente em condições de rede variáveis. A diretoria reconheceu a necessidade de aprimorar a qualidade do software e solicitou uma revisão nos Requisitos Não Funcionais (RNFs) que foram negligenciados. No contexto apresentado, assinale a ação capaz de garantir a qualidade do software:

- a) Realizar uma pesquisa de mercado para identificar as principais funcionalidades ausentes no aplicativo e integrá-las no menor tempo possível.
- b) Priorizar a adição de novos recursos ao aplicativo, assumindo que isso automaticamente resolverá os problemas de desempenho e usabilidade relatados pelos usuários.
- c) Implementar uma estratégia de marketing mais agressiva para melhorar a percepção do produto pelo mercado, desviando a atenção dos problemas de desempenho e usabilidade.
- d) Focar na otimização do código para melhorar o desempenho do aplicativo, assim como aprimorar a interface com o usuário, garantindo uma execução mais rápida em todas as condições de rede e melhor usabilidade para o usuário.
- e) Conduzir uma revisão focada nos Requisitos Não Funcionais, especialmente aqueles relacionados à usabilidade e ao desempenho em condições de rede variáveis, para identificar lacunas e oportunidades de melhoria.

7. (FGV / CGE-SC – 2023) A respeito do requisito não funcional, assinale V para a afirmativa verdadeira e F para a falsa.

- ( ) O sistema deve permitir que os usuários criem e editem seus perfis.
- ( ) O sistema deve estar disponível na Web e em dispositivos Android.
- ( ) O sistema deve ter um recurso de pesquisa que permita aos usuários encontrar informações sobre vendas.
- ( ) O sistema deve estar disponível 99,99% do tempo.
- ( ) O sistema deve exibir dados de vendas em tempo real.



As afirmativas são, respectivamente,

- a) V – F – V – F – V.
- b) V – V – V – F – V.
- c) F – V – F – V – F.
- d) V – F – V – V – V.
- e) V – F – F – F – V.

8. (FGV / AL-MA – 2023) A elicitação de requisitos é uma atividade central da engenharia de requisitos. Assinale a técnica de elicitação que extrai informações necessárias para construir um novo sistema a partir do código fonte de um sistema legado:

- a) arqueologia de sistemas.
- b) entrevista dos stakeholders.
- c) leitura baseada em perspectiva.
- d) mapas mentais.
- e) observação de campo.

9. (FGV / TCE-SP – 2023) A Equipe de Sistemas (ES) de um tribunal está trabalhando na especificação de requisitos para o Sistema de Protocolos (SisProt). O representante do cliente solicitou que o SisProt funcione 24 horas por dia, durante todo o ano, e explicou que será tolerada a interrupção do serviço prestado pelo SisProt por, no máximo, 1 hora por mês, seja para manutenções, seja por motivos extra sistema. O requisito solicitado pelo representante do cliente é categorizado como um requisito de:

- a) instabilidade;
- b) confiabilidade;
- c) manutenibilidade;
- d) tempo de resposta;
- e) responsabilidade (“accountability”).

10. (FGV / TCE-SP – 2023) O sistema SisBRAVO foi desenvolvido aderente ao preconizado na Lei nº 13.709/2018 – Lei Geral de Proteção de Dados (LGPD). O SisBRAVO solicita autorização para coleta de dados pessoais inseridos pelos usuários. Sendo assim, o SisBRAVO atende requisitos tipificados como:

- a) externos;
- b) organizacionais;
- c) de confiança;
- d) de confidencialidade;
- e) de adequação funcional.

11. (FGV / TCE-TO – 2022) A Equipe de Desenvolvimento de Soluções (EDS) recebeu a solicitação de que um dos campos utilizados para entrada de dados da aplicação Web em construção



apresente sugestões de palavras dinamicamente, conforme o usuário for digitando novos caracteres.

A EDS recebeu a solicitação de um requisito de:

- a) confiança;
- b) eficiência;
- c) desempenho;
- d) usabilidade;
- e) desenvolvimento.

**12. (FGV / TCE-TO – 2022)** Carlos é uma parte interessada em uma aplicação Web e solicitou à equipe de desenvolvimento uma funcionalidade capaz de emitir relatórios com cabeçalhos padronizados. Assim, os cabeçalhos devem ter cor de fundo, paleta de cores e tipografia, seguindo o padrão adotado em outros documentos emitidos pelo departamento responsável pela aplicação.

A solicitação de Carlos refere-se a um requisito:

- a) funcional regulador;
- b) não funcional organizacional;
- c) funcional do processo operacional;
- d) não funcional de usabilidade;
- e) funcional de desenvolvimento.

**13. (FGV / TCE-TO – 2022)** A Equipe de Desenvolvimento de Soluções (EDS) recebeu a solicitação de que um dos campos utilizados para entrada de dados da aplicação Web em construção apresente sugestões de palavras dinamicamente, conforme o usuário for digitando novos caracteres. A EDS recebeu a solicitação de um requisito de:

- a) confiança;
- b) eficiência;
- c) desempenho;
- d) usabilidade;
- e) desenvolvimento.

**14. (FGV / TCE-TO – 2022)** Carlos é uma parte interessada em uma aplicação Web e solicitou à equipe de desenvolvimento uma funcionalidade capaz de emitir relatórios com cabeçalhos padronizados. Assim, os cabeçalhos devem ter cor de fundo, paleta de cores e tipografia, seguindo o padrão adotado em outros documentos emitidos pelo departamento responsável pela aplicação:

A solicitação de Carlos refere-se a um requisito:





- a) funcional regulador;
- b) não funcional organizacional;
- c) funcional do processo operacional;
- d) não funcional de usabilidade;
- e) funcional de desenvolvimento.

**15. (FGV / TCE-TO – 2022)** As credenciais de acesso dos usuários de um aplicativo são armazenadas em um banco de dados e são utilizadas unicamente para acesso às funcionalidades do aplicativo. A equipe de desenvolvimento definiu como requisito não funcional que o sistema deve evitar que as senhas sejam obtidas por um invasor mesmo que o aplicativo ou banco de dados esteja comprometido.

Para implementar o requisito não funcional, um modo de proteger as senhas dos usuários é:

- a) ocultar o algoritmo utilizado para proteção das senhas;
- b) armazenar as senhas cifradas por meio de um algoritmo de chaves assimétricas;
- c) utilizar um algoritmo hash com salt antes de persistir as senhas no banco de dados;
- d) codificar um algoritmo próprio para cifrar as senhas com base em uma chave randômica segura;
- e) usar uma chave randômica gerada pelo aplicativo para cifrar as senhas por meio de um algoritmo de chave simétrica.

**16. (FGV / TCE-TO – 2022)** A Equipe de Tecnologia (ETi) de um tribunal de contas está levantando as necessidades para um novo sistema junto às partes interessadas. Uma das partes interessadas solicitou que o novo sistema seja fácil de usar, como requisito não funcional.

Para que o requisito não funcional “fácil de usar” seja objetivamente testado, a ETi deve considerar a métrica:

- a) eficiência;
- b) disponibilidade;
- c) tempo de treinamento;
- d) taxa de ocorrência de falhas;
- e) tempo de atualização de tela.

**17. (FGV / ALERJ – 2017)** O Analista de Sistemas Pedro está realizando um levantamento de requisitos por meio da prototipação. Sua intenção com esse protótipo é proporcionar uma visão geral do sistema com todas as suas funcionalidades, sem entrar em detalhes específicos de cada funcionalidade, de forma que a interface como um todo possa ser criticada pelos usuários. Nesse caso, o tipo de protótipo mais adequado é o (a):

- a) vertical;
- b) tridimensional;
- c) prototipação rápida;



- d) textual;
- e) horizontal.

**18. (FGV / BADESC – 2010)** Analise o fragmento a seguir:

"A base de dados deve ser protegida para acesso apenas de usuários autorizados".

O fragmento acima apresenta um exemplo do seguinte requisito:

- a) funcional.
- b) de usuário.
- c) de sistema.
- d) de domínio.
- e) não-funcional.

**19. (FGV / FIOCRUZ – 2010)** Sobre os processos de engenharia de requisitos, na elicitação e na análise ocorre total interação com os stakeholders no sistema, sendo o principal objetivo:

- a) a obtenção dos requisitos.
- b) a homologação do sistema.
- c) a elaboração do manual do usuário.
- d) a conversão de especificações em requisitos.
- e) a execução do estudo de viabilidade do sistema.

**20. (FGV / MEC – 2009)** Requisitos não-funcionais estão diretamente relacionados com a satisfação dos usuários. Assinale a alternativa que não indique um requisito não-funcional:

- a) O sistema de arquivos deve ser protegido, para acesso, apenas, de usuários autorizados.
- b) O software deve ser implementado usando os conceitos de orientação a objetos.
- c) O tempo de desenvolvimento do software não deve ultrapassar seis meses.
- d) O software poderá ser executado em plataforma windows e linux.
- e) O software deve emitir relatórios de vendas a cada quinze dias.

**21. (FGV / MEC – 2009)** As declarações de serviços que o sistema deve fornecer, de como ele deve reagir a entradas específicas ou se comportar em determinadas situações, são chamadas de requisitos:

- a) não-funcionais.
- b) de domínio.
- c) de sistema.
- d) funcionais.
- e) de usuário.



**22. (FGV / MEC – 2009)** Existem técnicas que são usadas na fase de levantamento de requisitos para coletar conhecimento dos usuários sobre os requisitos. Assinale a alternativa que indique apenas técnicas utilizadas na fase de levantamento de requisitos.

- a) JAD, WFMS, WBS, cenários e brainstorming.
- b) JAD, cenários, WFMS, questionários e intercepting.
- c) cenários, entrevistas, protótipos, workshop, brainstorming.
- d) leitura de documentos, protótipos, workshop, WBS e workflow.
- e) brainstorming, protótipos, workflow, leitura de documentos e intercepting.

**23. FGV / Senado Federal – 2008)** Entre as atividades listadas a seguir, uma não faz parte da Engenharia de Requisitos. Assinale-a.

- a) estudo de viabilidade.
- b) análise de risco.
- c) levantamento de necessidades do cliente.
- d) verificação.
- e) gerenciamento.



## GABARITO

- |    |         |     |         |     |         |
|----|---------|-----|---------|-----|---------|
| 1. | LETRA D | 9.  | LETRA B | 17. | LETRA E |
| 2. | LETRA C | 10. | LETRA A | 18. | LETRA E |
| 3. | LETRA D | 11. | LETRA D | 19. | LETRA A |
| 4. | LETRA D | 12. | LETRA B | 20. | LETRA E |
| 5. | LETRA B | 13. | LETRA D | 21. | LETRA D |
| 6. | LETRA E | 14. | LETRA B | 22. | LETRA C |
| 7. | LETRA C | 15. | LETRA C | 23. | LETRA B |
| 8. | LETRA A | 16. | LETRA C |     |         |



## LISTA DE QUESTÕES – DIVERSAS BANCAS

- 1. (CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento ágil decidiu utilizar Pontos de História (Story Points) para estimar o esforço necessário para implementar as funcionalidades descritas no seu Product Backlog, na forma de Histórias do Usuário. Considerando-se esse contexto, a técnica de Pontos de História ajuda a equipe a compreender o trabalho envolvido em cada história por:
  - a) alocar tarefas específicas da história do usuário aos membros da equipe.
  - b) classificar as histórias do usuário em categorias de prioridade, de alta a baixa.
  - c) permitir prever o tempo, em horas, que a equipe levará para completar uma história.
  - d) determinar o custo financeiro direto associado ao desenvolvimento de uma história do usuário.
  - e) estimar o esforço relativo necessário para implementar uma determinada história do usuário, comparando-o ao de outras histórias.
- 2. (CESGRANRIO / IPEA – 2024)** Um especialista em testes de software, com vasta experiência na criação de testes não funcionais, reuniu-se com sua equipe de desenvolvimento para avaliar se ainda havia requisitos não funcionais pendentes de teste em um novo software de e-commerce que estava sendo desenvolvido. Após ler a lista de requisitos identificados pela equipe como pendentes de teste, o especialista identificou um requisito não funcional ainda não testado. Com base no cenário apresentado, o requisito não funcional identificado foi o de verificar se:
  - a) a opção esqueci senha, ao ser pressionada pelo usuário, o direcionava para uma tela que solicitava o e-mail do usuário para recuperar tal senha.
  - b) o tempo de resposta para aprovar ou não alguma solicitação era de até 8 segundos.
  - c) o chatbot do sistema está apresentando o conteúdo esperado, a partir de informações fornecidas por algum usuário em um chat de ajuda acessado pelo software.
  - d) algum dos cupons de desconto de 30% para certos produtos escolhidos estava sendo aplicado corretamente.
  - e) os relatórios de venda do mês estavam sendo exportados no formato PDF e CSV.
- 3. (CESGRANRIO / IPEA – 2024)** A engenharia de requisitos pode ser dividida em gerenciamento de requisitos e desenvolvimento de requisitos, e este, por sua vez, pode ser subdividido em elicitação, análise, especificação e validação. Cada um desses quatro componentes do



desenvolvimento de requisitos possui um conjunto de boas práticas. Uma das boas práticas associadas à validação de requisitos é:

- a) priorizar os requisitos.
- b) registrar as regras de negócio.
- c) alocar requisitos aos subsistemas.
- d) definir critérios de aceitação do produto.
- e) especificar os atributos de qualidade do produto.

4. **(CESGRANRIO / IPEA – 2024)** No projeto de desenvolvimento de um determinado software, uma equipe foi designada para criar, exclusivamente, testes funcionais e não funcionais para esse software. Nesse cenário, é recomendado que a equipe se envolva nesse projeto na etapa de:

- a) requisitos
- b) produção
- c) manutenção
- d) homologação
- e) implementação

5. **(CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento de um software para gerência de finanças pessoais decidiu adotar uma abordagem ágil, utilizando Histórias do Usuário para capturar requisitos funcionais. Essa técnica tem como característica descrever as funcionalidades do software do ponto de vista do usuário final. Para assegurar uma compreensão comum entre todos os envolvidos no projeto, é fundamental que a equipe entenda os componentes de uma História do Usuário. Os três componentes principais de uma História do Usuário são:

- a) Cartão, Conversação e Confirmação.
- b) Classes, Métodos e Atributos.
- c) Entidade, Relacionamento e Atributo.
- d) Requisitos Funcionais, Requisitos Não Funcionais e Requisitos de Domínio.
- e) Diagrama de Caso de Uso, Diagrama de Atividades e Diagrama de Sequência.

6. **(CESGRANRIO / TRANSPETRO – 2023)** Os requisitos não funcionais podem ser divididos em três categorias: organizacionais, externos e de produto. Dentre os pertencentes à categoria organizacional estão os requisitos:

- a) legais
- b) operacionais
- c) reguladores
- d) de eficiência
- e) de usabilidade.



7. **(CESGRANRIO / TRANSPETRO – 2023)** Ao desenvolver protótipos de alta fidelidade de um software, é necessário:
- a) aplicar a técnica de paper prototyping.
  - b) realizar a primeira validação com algum usuário ou com algum cliente do protótipo criado.
  - c) definir o wireframe das telas do software sem o uso das cores da solução.
  - d) fornecer um layout básico que permita algumas validações iniciais de maneira digital.
  - e) oferecer protótipos de telas que representam de maneira fiel o que é esperado pelos usuários para que assim sejam codificadas.
8. **(VUNESP / TCM-SP – 2023)** O estudo de viabilidade constitui-se na fase inicial do desenvolvimento da especificação de requisitos do software e o resultado desse estudo deve, prioritariamente, compreender:
- a) uma planilha de custos detalhada de todas as atividades e compras a serem feitas no decorrer do desenvolvimento do software.
  - b) a recomendação para prosseguir, ou não, com os demais processos componentes da engenharia de requisitos.
  - c) uma classificação detalhada dos requisitos levantados, por ordem de prioridade e de tempo necessários para o desenvolvimento.
  - d) a definição de quais deverão ser os módulos de software desenvolvidos internamente e quais deverão ser adquiridos de terceiros.
  - e) a definição da equipe de desenvolvimento do software, bem como sua seleção, segundo critérios estabelecidos nesse estudo.
9. **(VUNESP / UFABC – 2023)** A Engenharia de Requisitos considera duas grandes vertentes: requisitos funcionais e requisitos não funcionais. Considerando os requisitos não funcionais, há uma classificação que os divide nas categorias de produto, organizacionais e externos, sendo certo que requisitos:
- a) operacionais inserem-se na categoria de requisitos de produto.
  - b) legais inserem-se na categoria de requisitos organizacionais.
  - c) ambientais inserem-se na categoria de requisitos externos.
  - d) de eficiência inserem-se na categoria de requisitos de produto.
  - e) éticos inserem-se na categoria de requisitos de produto.
10. **(CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento ágil decidiu utilizar Pontos de História (Story Points) para estimar o esforço necessário para implementar as funcionalidades descritas no seu Product Backlog, na forma de Histórias do Usuário. Considerando-se esse





contexto, a técnica de Pontos de História ajuda a equipe a compreender o trabalho envolvido em cada história por:

- a) alocar tarefas específicas da história do usuário aos membros da equipe.
- b) classificar as histórias do usuário em categorias de prioridade, de alta a baixa.
- c) permitir prever o tempo, em horas, que a equipe levará para completar uma história.
- d) determinar o custo financeiro direto associado ao desenvolvimento de uma história do usuário.
- e) estimar o esforço relativo necessário para implementar uma determinada história do usuário, comparando-o ao de outras histórias.

**11. (CESGRANRIO / IPEA – 2024)** Um especialista em testes de software, com vasta experiência na criação de testes não funcionais, reuniu-se com sua equipe de desenvolvimento para avaliar se ainda havia requisitos não funcionais pendentes de teste em um novo software de e-commerce que estava sendo desenvolvido. Após ler a lista de requisitos identificados pela equipe como pendentes de teste, o especialista identificou um requisito não funcional ainda não testado. Com base no cenário apresentado, o requisito não funcional identificado foi o de verificar se:

- a) a opção esqueci senha, ao ser pressionada pelo usuário, o direcionava para uma tela que solicitava o e-mail do usuário para recuperar tal senha.
- b) o tempo de resposta para aprovar ou não alguma solicitação era de até 8 segundos.
- c) o chatbot do sistema está apresentando o conteúdo esperado, a partir de informações fornecidas por algum usuário em um chat de ajuda acessado pelo software.
- d) algum dos cupons de desconto de 30% para certos produtos escolhidos estava sendo aplicado corretamente.
- e) os relatórios de venda do mês estavam sendo exportados no formato PDF e CSV.

**12. (CESGRANRIO / IPEA – 2024)** A engenharia de requisitos pode ser dividida em gerenciamento de requisitos e desenvolvimento de requisitos, e este, por sua vez, pode ser subdividido em elicitação, análise, especificação e validação. Cada um desses quatro componentes do desenvolvimento de requisitos possui um conjunto de boas práticas. Uma das boas práticas associadas à validação de requisitos é:

- a) priorizar os requisitos.
- b) registrar as regras de negócio.
- c) alocar requisitos aos subsistemas.



- d) definir critérios de aceitação do produto.
- e) especificar os atributos de qualidade do produto.

**13. (CESGRANRIO / IPEA – 2024)** No projeto de desenvolvimento de um determinado software, uma equipe foi designada para criar, exclusivamente, testes funcionais e não funcionais para esse software. Nesse cenário, é recomendado que a equipe se envolva nesse projeto na etapa de:

- a) requisitos
- b) produção
- c) manutenção
- d) homologação
- e) implementação

**14. (CESGRANRIO / Caixa – 2024)** Uma equipe de desenvolvimento de um software para gerência de finanças pessoais decidiu adotar uma abordagem ágil, utilizando Histórias do Usuário para capturar requisitos funcionais. Essa técnica tem como característica descrever as funcionalidades do software do ponto de vista do usuário final. Para assegurar uma compreensão comum entre todos os envolvidos no projeto, é fundamental que a equipe entenda os componentes de uma História do Usuário. Os três componentes principais de uma História do Usuário são:

- a) Cartão, Conversação e Confirmação.
- b) Classes, Métodos e Atributos.
- c) Entidade, Relacionamento e Atributo.
- d) Requisitos Funcionais, Requisitos Não Funcionais e Requisitos de Domínio.
- e) Diagrama de Caso de Uso, Diagrama de Atividades e Diagrama de Sequência.

**15. (CESGRANRIO / TRANSPETRO – 2023)** Os requisitos não funcionais podem ser divididos em três categorias: organizacionais, externos e de produto. Dentre os pertencentes à categoria organizacional estão os requisitos:

- a) legais
- b) operacionais
- c) reguladores
- d) de eficiência
- e) de usabilidade.

**16. (CESGRANRIO / TRANSPETRO – 2023)** Ao desenvolver protótipos de alta fidelidade de um software, é necessário:

- a) aplicar a técnica de paper prototyping.
- b) realizar a primeira validação com algum usuário ou com algum cliente do protótipo criado.
- c) definir o wireframe das telas do software sem o uso das cores da solução.



- d) fornecer um layout básico que permita algumas validações iniciais de maneira digital.
- e) oferecer protótipos de telas que representam de maneira fiel o que é esperado pelos usuários para que assim sejam codificadas.

**17.(VUNESP / TCM-SP – 2023)** O estudo de viabilidade constitui-se na fase inicial do desenvolvimento da especificação de requisitos do software e o resultado desse estudo deve, prioritariamente, compreender:

- a) uma planilha de custos detalhada de todas as atividades e compras a serem feitas no decorrer do desenvolvimento do software.
- b) a recomendação para prosseguir, ou não, com os demais processos componentes da engenharia de requisitos.
- c) uma classificação detalhada dos requisitos levantados, por ordem de prioridade e de tempo necessários para o desenvolvimento.
- d) a definição de quais deverão ser os módulos de software desenvolvidos internamente e quais deverão ser adquiridos de terceiros.
- e) a definição da equipe de desenvolvimento do software, bem como sua seleção, segundo critérios estabelecidos nesse estudo.

**18.(VUNESP / UFABC – 2023)** A Engenharia de Requisitos considera duas grandes vertentes: requisitos funcionais e requisitos não funcionais. Considerando os requisitos não funcionais, há uma classificação que os divide nas categorias de produto, organizacionais e externos, sendo certo que requisitos:

- a) operacionais inserem-se na categoria de requisitos de produto.
- b) legais inserem-se na categoria de requisitos organizacionais.
- c) ambientais inserem-se na categoria de requisitos externos.
- d) de eficiência inserem-se na categoria de requisitos de produto.
- e) éticos inserem-se na categoria de requisitos de produto.

**19.(IBFC / IBGE – 2021)** A etapa de levantamento de requisitos é composta por diversas técnicas que visam obter do cliente as informações necessárias para desenvolver o projeto do sistema de informação. Sobre essas técnicas, analise as afirmativas abaixo, dê valores Verdadeiro (V) ou Falso (F).

- ( ) Entrevistas não estruturadas: Informal ou sem agenda pré-definida.
- ( ) Brainstorming: Reunião com várias pessoas onde todos discutem um tema central.
- ( ) Prototipagem: Desenvolvimento de um modelo que simulará o sistema real.

- a) F - F - F



- b) F - F - V
- c) V - F - V
- d) V - V - F
- e) V - V - V

**20. (IDIB / CRF – MS – 2021)** Em qual etapa do processo de desenvolvimento de requisitos de software mais comumente costuma acontecer a identificação de gaps nos requisitos ou a identificação de requisitos desnecessários, conforme eles se relacionam com o escopo definido?

- a) Elicitação de requisitos.
- b) Análise de requisitos.
- c) Especificação de requisitos.
- d) Validação de requisitos.

**21. (IDIB / CRF - MS – 2021)** "Uma descrição de uma propriedade ou característica que um sistema deve exibir ou uma restrição que ele deve respeitar". Tal definição se adequa a qual tipo de requisito de informação em um processo de software?

- a) Requisito Funcional.
- b) Requisito Não-Funcional.
- c) Atributo de Qualidade.
- d) Requisito do Usuário.

**22. (VUNESP / EBSERH – 2020)** Considerando as técnicas utilizadas para a avaliação de requisitos, é correto afirmar que, na:

- a) facilidade de verificação, deve-se verificar se não há requisitos conflitantes entre si.
- b) verificação de consistência, deve-se verificar se os requisitos podem ser implementados, considerando a tecnologia existente.
- c) verificação de realismo, deve-se verificar se todas as funções e restrições planejadas estão contempladas.
- d) verificação de validade, deve-se verificar se não há requisitos conflitantes entre si.
- e) verificação de completeza, deve-se verificar se todas as funções e restrições planejadas estão contempladas.

**23. (VUNESP / EBSERH – 2020)** Na engenharia de requisitos, um fator importante são os requisitos não funcionais, que se classificam em organizacionais, de produto e externos. Os requisitos

- a) de produto têm origem em políticas e procedimentos da organização do cliente.
- b) de produto compreendem fatores oriundos de fatores externos ao sistema e seu processo de desenvolvimento.
- c) externos especificam o comportamento do produto, tais como o desempenho e a memória requerida.
- d) organizacionais têm origem em políticas e procedimentos da organização do cliente.



e) organizacionais especificam o comportamento do produto, tais como o desempenho e a memória requerida.

**24. (COPESE - UFPI / ALEPI – 2020)** Um técnico de TI da ALEPI que gerencia uma equipe de desenvolvimento de software na Assembleia, eventualmente necessita fazer levantamento de requisitos da aplicação que está sendo desenvolvida. Sobre os Requisitos de Software, considere as seguintes afirmativas:

I. Requisitos funcionais são aqueles que definem parte da funcionalidade do sistema e podem ser categorizados em três tipos: requisitos de produtos, requisitos organizacionais e requisitos externos.

II. Requisitos não-funcionais dizem respeito a restrições, aspectos de desempenho, interfaces com o usuário, confiabilidade, segurança, portabilidade e padrões.

III. Requisitos organizacionais estão relacionados às metas da empresa, suas políticas estratégicas adotadas, assim como assuntos relacionados aos empregados da empresa com seus respectivos objetivos.

IV. Requisitos de produto estão relacionados as restrições impostas por fatores externos ao sistema tais como restrições de interoperabilidade, éticas e legais.

Marque a opção que corresponde somente às afirmativas verdadeiras.

- a) Apenas I, III e IV.
- b) Apenas I, II e III.
- c) Apenas I e III.
- d) Apenas II e III.
- e) Apenas II, III e IV.

**25. (FAFIPA / Prefeitura de Arapongas - PR – 2020)** A Engenharia de Requisitos é um termo cunhado para descrever as atividades relacionadas à investigação e definição de escopo de um sistema de software, ou seja, trata-se do processo de descobrir, analisar, documentar e verificar as funções e restrições do sistema. Para auxiliar o levantamento de requisitos, existe um conjunto de técnicas de levantamento de dados que podem ser aplicadas em conjunto ou isoladamente, a depender das características do projeto. Assinale a alternativa que apresenta somente técnicas para descoberta de requisitos:

- a) Sprint; Caso de Uso; Etnografia.
- b) Entrevista; Caso de Uso; Etnografia.
- c) Sprint; Refatoração; Etnografia.
- d) JAD; Refatoração; Etnografia.
- e) Entrevista; JAD; Refatoração.



**26. (IBFC / EBSE RH – 2020)** Requisitos são as bases para todo projeto, definindo o que as partes interessadas de um novo sistema necessitam e também o que o sistema deve fazer para satisfazer as suas necessidades. Antes do processo, propriamente dito, da Engenharia de Requisitos deve-se ter:

- a) entrevistas e questionários com os usuários
- b) a documentação dos requisitos
- c) a revisão dos requisitos funcionais e não-funcionais
- d) os estudos de viabilidade técnica/financeira
- e) a revisão dos requisitos pelos usuários

**27. (INSTITUTO AOCP / Prefeitura de Betim - MG – 2020)** A engenharia de requisitos estabelece uma ponte entre o projeto e a construção do software. Assinale a alternativa que representa a etapa na qual as metas de negócio são estabelecidas.

- a) Levantamento.
- b) Concepção.
- c) Negociação.
- d) Revisão.
- e) Gestão.

**28. (CESGRANRIO / TRANSPETRO – 2018)** Os requisitos não funcionais são requisitos que não estão diretamente relacionados com o que o sistema é capaz de fazer para atender aos seus usuários. Eles podem estar relacionados a algumas características, propriedades ou restrições como confiabilidade, ambiente de operação, aspectos legais e outros. Alguns autores os classificam em requisitos de produto, organizacionais e externos.

Os requisitos de produto ainda são classificados como de usabilidade, confiança, proteção e:

- a) ambientais
- b) operacionais
- c) segurança
- d) éticos
- e) eficiência

**29. (CESGRANRIO / TRANSPETRO – 2018)** Ao realizar uma especificação de requisitos, um analista resolveu utilizar a técnicas QFD. Nessa técnica, os requisitos esperados têm como característica ser:

- a) explícitos e causar insatisfação significativa se não implementados.
- b) implícitos e causar insatisfação significativa se não implementados.
- c) explícitos e causar satisfação se implementados.
- d) implícitos e causar satisfação se implementados.
- e) desconhecidos e causar satisfação significativa se implementados.



**30. (CESGRANRIO / TRANSPETRO – 2018)** A técnica de etnografia para o levantamento de requisitos implica a:

- a) imersão do analista no ambiente de trabalho onde o sistema vai ser usado.
- b) imersão do usuário no ambiente de desenvolvimento.
- c) realização de reuniões de levantamento de requisitos com grupos de usuários de um mesmo tipo.
- d) realização de reuniões de levantamento de requisitos com grupos de usuários de tipos diferentes.
- e) realização de reuniões de levantamento no ambiente de trabalho do usuário.

**31. (CESGRANRIO / TRANSPETRO – 2018)** Há autores que classificam os requisitos não funcionais de um sistema de informação em requisitos de produto, requisitos organizacionais e requisitos externos. Cada uma dessas categorias também sofre subdivisões. Considerando essa informação e o desenvolvimento de um sistema acadêmico, são pares que estabelecem a correspondência (categoria de requisito; exemplo), EXCETO:

- a) (eficiência; os relatórios de notas devem ser impressos em menos de cinco segundos).
- b) (segurança; o sistema deve prover recursos para registro de ponto do professor por reconhecimento de impressão digital).
- c) (ética; cada trabalho proposto possuirá um único prazo para todos os alunos do curso).
- d) (usabilidade; o professor lança as faltas dos alunos das turmas nas quais leciona).
- e) (ambiental; os servidores devem trabalhar entre 40% e 55% de umidade relativa do ar e 5° a 15° de temperatura).

**32. (CESGRANRIO / Banco da Amazônia – 2018)** Requisitos existem em vários níveis de abstração. Um desses níveis é conhecido como “requisitos de negócio”, os quais:

- a) definem a visão e o escopo que influenciam os requisitos de usuário.
- b) são dependentes dos requisitos funcionais para serem levantados.
- c) são descobertos depois e a partir dos requisitos de usuário.
- d) são registrados na forma da Especificação de Requisitos de Software.
- e) influenciam e são registrados na forma de regras de negócio.

**33. (CESGRANRIO / Petrobras – 2018)** O processo de desenvolvimento de requisitos pode ser dividido em 4 etapas: elicitação, análise, especificação e validação. Qual prática da Engenharia de Requisitos está associada à etapa de análise?

- a) Criar protótipos.





- b) Identificar casos de uso.
- c) Inspeccionar documentos de requisitos.
- d) Definir critérios de aceitação.
- e) Registrar regras de negócio.

**34. (CESGRANRIO / Petrobras – 2018)** Um analista em treinamento apresenta a seguinte descrição para alguns requisitos de um sistema on-line para venda de bilhetes de uma rede de cinemas em que cada cinema apresenta mais de um filme, e cada filme é apresentado em mais de um cinema com lugares marcados. Quando o usuário entrar no site da rede, um menu é apresentado com os filmes e cinemas disponíveis. O usuário seleciona o filme a que deseja assistir e entra com o número de cartão de crédito. A validade do cartão é verificada e o bilhete é emitido. O sistema encerra a operação com a mensagem: obrigado, divirta-se. Embora haja falhas nessa descrição ainda incipiente, o instrutor solicitou que o analista a estendesse com exemplos de possíveis requisitos não funcionais para o sistema.

Qual é a sentença que NÃO atende à solicitação do instrutor?

- a) O tempo máximo de espera do sistema para a inserção do número do cartão de crédito deve ser maior do que oito segundos e menor do que 15 segundos.
- b) O sistema deve estar disponível por 99,5% do tempo durante as 24 horas do dia, sete dias por semana.
- c) O formato do bilhete deve seguir o padrão definido na legislação municipal pertinente.
- d) O usuário deve selecionar o cinema em que deseja assistir ao filme.
- e) O sistema deve ser compatível com operação em smartphome.

**35. (FAURGS / TJ-RS – 2018)** Requisitos não funcionais – como o nome sugere – são requisitos que não estão diretamente relacionados com os serviços específicos oferecidos pelo sistema a seus usuários. Podem ser provenientes das características requeridas para o software, da organização que desenvolve o software ou de fontes externas. Os requisitos não funcionais que especificam ou restringem o comportamento do software – como por exemplo o seu desempenho, seus requisitos de proteção, seus requisitos de usabilidade e a taxa aceitável de falhas – são denominados requisitos:

- a) organizacionais.
- b) de produto.
- c) externos.
- d) éticos.
- e) ambientais.



**36. (FAURGS / TJ-RS – 2018)** Técnicas de descoberta de requisitos (às vezes chamada de elicitaco de requisitos) é o processo de reunir informaões sobre o sistema requerido e os sistemas existentes e separar dessas informaões os requisitos do usurio e de sistema; o uso destas tcnicas faz parte da maioria dos processos de engenharia de requisitos. \_\_\_\_\_ é o nome dado a uma tcnica em que o stakeholder responde a um conjunto predefinido de perguntas sobre o sistema usado no momento e sobre o sistema que ser desenvolvido; os requisitos surgem a partir das respostas a essas perguntas.

A alternativa que contm o termo que completa corretamente a lacuna do texto acima é:

- a) Entrevista fechada.
- b) Entrevista aberta.
- c) Etnografia.
- d) Cenrios.
- e) Casos de uso.

**37. (FAURGS / TJ-RS – 2018)** Qual alternativa abaixo apresenta um requisito funcional de software?

- a) A base de dados deve ser protegida para acesso apenas a usurios autorizados.
- b) O tempo de resposta do sistema no deve ultrapassar 30 segundos..
- c) O software deve ser operacionalizado no Sistema Operacional Windows.
- d) O software deve emitir relatrios de vendas.
- e) O tempo de desenvolvimento no deve ultrapassar trs meses.

**38. (FAURGS / TJ-RS – 2018)** Considerando que, durante o processo de validao de requisitos, estes so submetidos a diferentes tipos de verificao, assinale a alternativa cuja verificao indica que no documento de requisitos no existem descriões diferentes para uma mesma funo do sistema.

- a) Verificao de consistncia.
- b) Verificao de completude.
- c) Verificaões de realismo.
- d) Verificaões de validade.
- e) Prototipao.

**39. (UFG / SANEAGO – 2017)** Com relao à qualidade de software, é um exemplo de requisito de software no funcional:

- a) calcular o valor do desconto conforme o perfil do cliente.
- b) processar at 100 pedidos por segundo.
- c) produzir o software em at dois anos e ter custo inferior a R\$100.000,00.
- d) limitar o cadastro de usurios ao Departamento de Pessoal.



**40.(UFG / SANEAGO – 2017)** Alguns usuários estão insatisfeitos com um software. Uma investigação revelou que a origem da insatisfação decorre de uma omissão (requisito não especificado) na especificação de requisitos de software. Que atividade da engenharia de requisitos precisa ser revista para evitar problemas semelhantes?

- a) Análise de viabilidade.
- b) Validação.
- c) Construção.
- d) Elaboração do Termo de Abertura.

**41.(UFG / SANEAGO – 2017)** São atributos de requisito de software:

- a) custo (para implementar) e complexidade ciclomática.
- b) prioridade e linguagem de implementação.
- c) complexidade ciclomática e risco.
- d) risco e identificador.

**42.(UFG / SANEAGO – 2017)** Uma Engenharia de Requisitos (ER) bem estruturada garante qualidade, confiabilidade e integridade ao produto de software a ser desenvolvido. O conjunto de atributos que evidencia o esforço necessário para fazer modificações especificadas no software é uma característica de:

- a) portabilidade.
- b) confiabilidade.
- c) manutenibilidade.
- d) eficiência.

**43. (IBFC / EBSERH – 2017)** Quanto aos vários tipos de requisitos assinala, das alternativas abaixo, a única que NÃO identifica corretamente um clássico requisito não-funcional:

- a) requisito de implementação da arquitetura do sistema
- b) requisitos de funcionalidades do sistema
- c) requisito de interoperabilidade da arquitetura do sistema
- d) requisitos de confiabilidade da arquitetura do sistema
- e) requisitos de portabilidade da arquitetura do sistema

**44.(IBFC / EBSERH – 2017)** A Análise de Requisitos é a primeira fase de desenvolvimento de software dividido em Requisitos funcionais e Requisitos não-funcionais. Os Requisitos não-funcionais possuem vários tipos diferentes de classificação tais como:

- (1) Requisitos de confiabilidade.
- (2) Requisitos de produtos.
- (3) Requisitos éticos.
- (4) Requisitos de portabilidade.



- a) da relação apresentada existem somente o 2, 3 e 4
- b) da relação apresentada existem somente o 1, 3 e 4
- c) da relação apresentada existem somente o 1, 2 e 4
- d) da relação apresentada existem somente o 1, 2 e 3
- e) da relação apresentada existem todos.

**45. (CESGRANRIO / IBGE – 2016)** Um dos objetivos da disciplina de requisitos é:

- a) criar um esboço inicial da arquitetura do sistema a ser desenvolvido.
- b) adaptar e configurar o processo de desenvolvimento de modo a atender às especificidades do sistema a ser desenvolvido.
- c) fornecer uma base para estimar o custo e o tempo de desenvolvimento de um sistema.
- d) assegurar que os clientes, os usuários e os desenvolvedores tenham um entendimento comum da organização na qual um sistema será implantado.
- e) entender a estrutura e a dinâmica da organização na qual um sistema será implantado.

**46. (CESGRANRIO / IBGE – 2016)** Um dos objetivos da disciplina de requisitos é:

- a) criar um esboço inicial da arquitetura do sistema a ser desenvolvido.
- b) adaptar e configurar o processo de desenvolvimento de modo a atender às especificidades do sistema a ser desenvolvido.
- c) fornecer uma base para estimar o custo e o tempo de desenvolvimento de um sistema.
- d) assegurar que os clientes, os usuários e os desenvolvedores tenham um entendimento comum da organização na qual um sistema será implantado.
- e) entender a estrutura e a dinâmica da organização na qual um sistema será implantado.

**47. (IBFC / MGS – 2015)** A definição: "descrevem as funcionalidades que se espera que o sistema disponibilize, de uma forma completa e consistente. É aquilo que o usuário espera que o sistema ofereça, atendendo aos propósitos para qual o sistema será desenvolvido.", corresponde tipicamente aos:

- a) Requisitos Funcionais.
- b) Requisitos Externos.
- c) Requisitos não-Funcionais.
- d) Requisitos da Aplicação.



**48. (CESGRANRIO / IBGE – 2014)** Solicitado para fazer o levantamento dos requisitos para um novo software a ser desenvolvido, um analista de sistemas identificou a necessidade de descobrir todos aqueles que se beneficiariam de forma direta ou indireta do sistema a ser desenvolvido.

Essas pessoas são conhecidas como:

- a) clientes
- b) partes interessada
- c) patrocinadores
- d) usuários
- e) usuários finais

**49. (CESGRANRIO / IBGE – 2014)** Solicitado para fazer o levantamento dos requisitos para um novo software a ser desenvolvido, um analista de sistemas identificou a necessidade de descobrir todos aqueles que se beneficiariam de forma direta ou indireta do sistema a ser desenvolvido. Essas pessoas são conhecidas como:

- a) clientes
- b) partes interessadas
- c) patrocinadores
- d) usuários
- e) usuários finais.

**50. (CESGRANRIO / Petroquímica SUAPE – 2012)** Na modelagem de sistemas, há um conjunto de atividades denominado Engenharia de Requisitos. Nesse contexto, o propósito da validação de requisitos é:

- a) assegurar que os requisitos menos arriscados sejam considerados inicialmente no desenvolvimento.
- b) assegurar que o produto de software sendo especificado está em conformidade com as necessidades dos usuários.
- c) aumentar o nível de abstração na comunicação entre especialistas do domínio e engenheiros de requisitos.
- d) definir formas de rastrear os requisitos a partir dos correspondentes artefatos de modelagem e de código eventualmente existentes.
- e) maximizar o reuso de requisitos voláteis identificados em outros sistemas.

**51. (CESGRANRIO / Petroquímica SUAPE – 2012)** Diversas técnicas são utilizadas na modelagem de sistemas de software para especificar requisitos. Duas dessas técnicas são: a com base em



casos de uso (use cases) e a com base em histórias de usuário (user stories). Essas técnicas, aplicáveis a um sistema de software, são recomendadas para:

- a) descrever a decomposição funcional recursiva desse sistema.
- b) descrever aspectos de implementação e da interface gráfica desse sistema.
- c) descrever funcionalidades desse sistema, na perspectiva das necessidades dos usuários.
- d) descrever requisitos não funcionais (tais como segurança e desempenho) desse sistema.
- e) eliminar os requisitos voláteis identificados para esse sistema.

**52. (CESGRANRIO / CAIXA – 2012)** Um dos objetivos das revisões técnicas formais de software é:

- a) realizar uma única reunião ao final do projeto para avaliar se o software foi bem construído.
- b) realizar reuniões com os clientes para descobrir o que deve ser feito.
- c) documentar os requisitos elicitados.
- d) garantir que o software não possui erros.
- e) garantir que o software atende aos requisitos especificados.

**53. (CESGRANRIO / Petrobras – 2012)** Em um projeto de software, o documento de requisitos de software é utilizado por vários usuários, cada um buscando no documento uma ou mais aplicações. Os Engenheiros de Sistemas buscam, principalmente,

- a) desenvolver testes de validação para o sistema
- b) entender qual sistema deve ser desenvolvido.
- c) entender o sistema e as relações entre suas partes
- d) especificar os requisitos do sistema.
- e) planejar o processo de desenvolvimento do sistema.

**54. (CESGRANRIO / Petrobras – 2012)** Ao determinar os requisitos de um projeto, é necessário levantar os requisitos funcionais e os não funcionais. Os requisitos não funcionais:

- a) definem com detalhes exatamente o que deve ser implementado.
- b) definem explicitamente as funções que o sistema não deve executar.
- c) indicam os serviços que o sistema deve prestar.
- d) representam restrições aos serviços oferecidos pelo sistema.
- e) são descrições de que serviços o sistema deve fornecer aos usuários.

**55. (CESGRANRIO / LIQUIGÁS – 2012)** O processo que determina se as especificações dos requisitos são consistentes com suas definições é chamado de Validação dos Requisitos. Dentre as técnicas automatizadas usadas nesse processo, citam-se os(as):

- a) cenários
- b) revisões
- c) entrevistas
- d) prototipações



e) provas matemáticas

**56. (CESGRANRIO / TRANSPETRO – 2012)** Conversando sobre o sistema de estoque, o usuário informou ao analista que tal sistema deverá ter interface gráfica e gerar, todos os dias, a lista de produtos perecíveis no prazo de validade. Essas informações dadas pelo usuário, referem-se, respectivamente, às seguintes definições:

- a) característica de qualidade e requisito não funcional
- b) requisito funcional e requisito de eficiência
- c) requisito não funcional e requisito funcional
- d) requisito funcional e característica de qualidade
- e) requisito de usabilidade e requisito de produto

**57. (CESGRANRIO / CHEST – 2012)** O processo de engenharia de requisitos engloba todas as atividades necessárias para criar e manter um documento de requisitos do sistema e compreender os elementos de negócio que serão atendidos pelo software a ser desenvolvido, pertencendo a uma sequência lógica de atividades que culminam em um documento de requisitos correto que inclui todas as necessidades do cliente. O primeiro passo necessário no processo de criação desse documento de requisitos deve ser a(o):

- a) análise de viabilidade
- b) prototipagem do sistema
- c) geração de casos de teste
- d) design da aplicação
- e) documento de gerenciamento de mudanças

**58. (CESGRANRIO / BR DISTRIBUIDORA – 2011)** Ao determinar um conjunto inicial de requisitos funcionais para um sistema Web de controle de pedidos feitos por postos de revenda de combustível, um analista inexperiente de uma empresa distribuidora chegou aos seguintes requisitos:

- I - O sistema deverá permitir ao representante do posto solicitar uma entrega de combustíveis.
- II - O sistema deverá utilizar JAVA.
- III - O sistema deverá suportar pedidos em português e espanhol.

É(São) requisito(s) funcional(is) APENAS o(s) descrito(s) em

- a) I
- b) II
- c) III
- d) I e II
- e) II e III





**59. (CESGRANRIO / FINEP – 2011)** Uma equipe de analistas está entrevistando gerentes de área para levantar os requisitos do novo sistema de reservas de uma companhia aérea. Considere as afirmativas sobre os requisitos levantados.

I - O usuário poderá fornecer um roteiro com múltiplos pontos de parada.

II - O total a ser pago deverá ser expresso na moeda escolhida pelo usuário.

III - As trocas de informações com os sistemas das empresas coligadas (hotéis, locadora de veículos, etc.) são feitas através de Web Services.

É(São) requisito(s) funcional(ais) o que é apresentado em.

- a) I, apenas.
- b) III, apenas.
- c) I e II, apenas.
- d) II e III, apenas.
- e) I, II e III.

**60. (CESGRANRIO / Petrobras – 2011)** Na Engenharia de Software, os requisitos que descrevem o comportamento externo do sistema, estabelecendo uma descrição detalhada das funções, dos serviços e das restrições operacionais do referido sistema, são os requisitos:

- a) funcionais
- b) externos
- c) de sistema
- d) do usuário
- e) não funcionais

**61. (CESGRANRIO / Petrobras – 2010)** Um analista de sistemas levantou a lista de requisitos de um sistema de contas a pagar (SCP).

R<sub>1</sub> = O contador cadastra no SCP uma ou várias contas.

R<sub>2</sub> = Diariamente o contador visualiza na tela as contas que devem ser pagas no dia.

R<sub>3</sub> = Um relatório é gerado contendo os pagamentos realizados em um dado mês.

R<sub>4</sub> = O contador deve poder acessar o SCP através de um navegador Internet padrão.

R<sub>5</sub> = O SCP deve estar disponível durante o horário de expediente.

Como são classificados estes requisitos?

- a) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>, R<sub>3</sub>, R<sub>4</sub>, R<sub>5</sub>  
Requisitos Não Funcionais - Nenhum
- b) Requisitos Funcionais - R<sub>1</sub>, R<sub>2</sub>  
Requisitos Não Funcionais - R<sub>3</sub>, R<sub>4</sub>, R<sub>5</sub>



c) Requisitos Funcionais - R1,R2,R3  
Requisitos Não Funcionais - R4,R5

d) Requisitos Funcionais - R1,R2,R3,R4  
Requisitos Não Funcionais - R5

e) Requisitos Funcionais - R1,R2,R4  
Requisitos Não Funcionais - R3,R5

**62. (CESGRANRIO / ELETROBRAS – 2010)** Uma equipe de desenvolvimento de sistemas foi contratada para confeccionar o software de controle de voo de uma nova aeronave. Sabendo-se que esse assunto é novo para os membros da equipe, a primeira ação a ser tomada, no contexto de levantamento de requisitos, é

- a) determinar as necessidades iniciais do cliente pela aplicação de questionários.
- b) realizar o modelo de negócios, usando diagramas UML para descrever os processos de negócio do cliente.
- c) Aumentar o máximo possível o nível de compreensão da equipe sobre o campo de aplicação.
- d) criar rapidamente um protótipo descartável do produto, com base nas entrevistas preliminares.
- e) verificar a viabilidade das especificações não funcionais exigidas pelo cliente.

**63. (CESGRANRIO / ELETROBRAS – 2010)** Para superar as dificuldades encontradas na execução do levantamento de requisitos de sistemas, uma empresa estuda as vantagens e as desvantagens de diferentes técnicas. Qual a técnica de levantamento de requisitos, baseada na observação, em que o analista se insere no ambiente de trabalho no qual o sistema será utilizado, para compreender a política organizacional e a cultura de trabalho, com o objetivo de familiarizar-se com o negócio e sua história?

- a) Workshop.
- b) JAD.
- c) Entrevista.
- d) Etnografia.
- e) Brainstorming.

**64. (CESGRANRIO / ELETROBRAS – 2010)** Uma fábrica de software recomenda que a documentação de especificação funcional de um sistema deve ser clara para o cliente e detalhada para o desenvolvedor, estabelecendo um contrato entre eles. Documentos de especificação funcional têm como característica:



- a) apresentar os diagramas de interação relacionados aos requisitos contidos na especificação.
- b) conter os requisitos não funcionais pertinentes ao problema a ser resolvido.
- c) conter instruções detalhadas sobre o que o sistema deve fazer e como ele deve ser implementado.
- d) ser descrito em uma linguagem matemática formal, facilitando o entendimento do cliente que, geralmente, não é um especialista técnico.
- e) definir os recursos responsáveis por implementar as funcionalidades.

**65. (CESGRANRIO / ELETROBRAS – 2010)** As técnicas de identificação de requisitos de sistemas possuem características apropriadas a cada situação. Nesse contexto, analise as afirmações sobre as técnicas a seguir, considerando que a abordagem baseada em

- I - Workshop de Requisitos utiliza momentos de descontração como forma de dinamizar o trabalho em equipe;
- II - Cenários utiliza exemplos práticos descritivos do comportamento de um sistema;
- III - Entrevistas e Questionários mostra-se inadequada na fase inicial de obtenção de dados.

Está correto o que se afirma em.

- a) I, apenas.
- b) I e II, apenas.
- c) I e III, apenas.
- d) II e III, apenas.
- e) I, II e III.

**66. (CESGRANRIO / Petrobras – 2010)** No contexto da engenharia de software, o processo conhecido como engenharia de requisitos permite ao engenheiro de software

- a) eliminar a volatilidade dos requisitos que foram levantados para o produto de software.
- b) postergar a definição do escopo inicial do sistema para etapas posteriores do desenvolvimento.
- c) realizar a Implantação da Função de Qualidade (IFQ), que corresponde a maximizar a qualidade do processo de software utilizado no desenvolvimento do produto de software já implantado.
- d) remover as dependências entre os requisitos do sistema e as regras de negócio.
- e) usar o modelo de casos de uso para especificar os requisitos funcionais do produto de software.

**67. (CESGRANRIO / IBGE – 2010)** Com o objetivo de minimizar os problemas enfrentados e melhorar o processo de engenharia de requisitos, um engenheiro de requisitos decidiu elencar uma série de medidas que poderá empregar em seus futuros projetos, tais como:



I - aplicar a técnica de IFQ (Implantação da Função de Qualidade) que permite coletar os requisitos excitantes, os quais refletem características que vão além das expectativas do cliente e mostram ser muito satisfatórios quando presentes;

II - utilizar tabelas de rastreamento que relacionam os requisitos identificados a um ou mais aspectos do sistema;

III - utilizar casos de uso para fazer uma coleta iterativa de requisitos, uma vez que o processo de levantamento de requisitos é uma atividade evolutiva.

Está(ão) correta(s) a(s) medida(s)

- a) I, apenas.
- b) II, apenas.
- c) III, apenas.
- d) I e II, apenas.
- e) I, II e III.

**68. (ESAF / CVM – 2010)** Assinale a opção correta.

- a) Gestão de requisitos preocupa-se com a documentação, atualização e controle de stakeholders envolvidos na fase de identificação da demanda.
- b) Engenharia de requisitos compreende: identificar, analisar, especificar e definir as necessidades de negócio que um aplicativo deve prover para solução do problema levantado.
- c) Engenharia de requisitos compreende: planejar, especificar e desenvolver as necessidades de negócio que um aplicativo deve prover para minimização dos problemas levantados.
- d) Engenharia de requisitos compreende: identificar, analisar, programar e testar os programas das necessidades de solução de problemas que um negócio deve prover para satisfazer usuários.
- e) Gestão de requisitos preocupa-se com a documentação, direcionamento, controle de definição e acesso aos requisitos levantados na fase de planejamento de escopo.

**69. (ESAF / MPOG – 2010)** As áreas de esforços da Análise de Requisitos são:

- a) reconhecimento dos objetivos, avaliação e controle, modelagem, estruturação e revisão.
- b) reconhecimento do problema, avaliação e síntese, modelagem, especificação e revisão.
- c) reengenharia, planejamento, avaliação e controle, modelagem e conclusão.



d) reconhecimento do problema, análise e síntese, reengenharia, especificação e análise de resultados.

e) reconhecimento do problema, modelagem, especificação de entidades, estruturação e revisão.

**70. (CESGRANRIO / TJ-RO – 2008)** Considere os quatro requisitos registrados em um projeto de uma aplicação para a Internet apresentados a seguir.

I - O tempo de resposta máximo do sistema a qualquer ação do usuário deve ser de 5s.

II - Clientes que tenham pago as últimas cinco compras à vista têm direito a um desconto não cumulativo de 10% na próxima compra.

III - A interface com o usuário deve ser organizada em abas e menus.

IV - Se o produto possuir uma quantidade máxima permitida por compra, esse limite deve ser imposto pelo sistema durante uma compra.

São tipicamente classificados como requisitos funcionais APENAS os requisitos.

- a) I e II
- b) I e III
- c) II e III
- d) II e IV
- e) III e IV

**71. (CESGRANRIO / Petrobras – 2008)** Analise as afirmativas a seguir, sobre requisitos em projetos de software.

I - O rastreamento de requisitos é de grande importância para conduzir análises de impacto quando há mudanças em requisitos.

II - O acrônimo FURPS+ se refere aos requisitos não funcionais das categorias de Feasibility, Usability, Reliability, Performance e Supportability.

III - Um requisito pode conter, além da especificação, atributos que sirvam ao seu gerenciamento.

IV - Casos de uso são descrições da interação entre um ator e o sistema e, portanto, especificam apenas requisitos funcionais.

Estão corretas APENAS as afirmativas.



- a) I e II
- b) I e III
- c) II e III
- d) II e IV
- e) III e IV

**72. (CESGRANRIO / TJ-RO – 2008)** Durante as atividades de Requisitos em um projeto de desenvolvimento de software, são realizadas entrevistas com clientes (usuários e stakeholders, no papel de entrevistados) com o objetivo de levantar suas necessidades e validar as características propostas para o software a ser desenvolvido. Os analistas, no papel de entrevistadores, em geral utilizam dois tipos de perguntas durante as entrevistas: perguntas livres de contexto e perguntas no contexto da solução. Sobre o tema, assinale a afirmativa correta.

- a) Perguntas livres de contexto proporcionam ao analista um entendimento do problema a ser resolvido pelo sistema sem influenciar o entrevistado com detalhes de uma solução que o analista já tenha em mente, baseada em experiências prévias, que pode não ser a mais adequada para o projeto.
- b) Perguntas no contexto da solução devem ser feitas antes das perguntas livres de contexto, de forma que seja possível obter um entendimento inicial do escopo do projeto e testar se a solução proposta coincide com as expectativas dos clientes.
- c) Perguntas no contexto da solução estão relacionadas à identificação das necessidades dos clientes, enquanto perguntas livres de contexto estão relacionadas às características do software a ser desenvolvido.
- d) Um exemplo de pergunta no contexto da solução é "Quem são os usuários (do software a ser desenvolvido)?".
- e) É consenso nos dias atuais que uma alternativa vantajosa às entrevistas é a distribuição de questionários aos clientes, com perguntas dos dois tipos, o que possibilita colher mais informações em menor tempo e comparar as respostas para detectar inconsistências.

**73. (CESGRANRIO / Petrobras – 2006)** Sobre a Análise e o Gerenciamento de Requisitos, é FALSO afirmar que:

- a) quanto mais tarde for identificado um problema na análise de requisitos, maior será o custo com o retrabalho.
- b) a elicitação é o processo de identificação e entendimento das necessidades e restrições dos usuários, enquanto que a especificação é o processo de formalização das necessidades e restrições dos usuários em requisitos funcionais de software.



c) na análise de requisitos o cliente utiliza as melhores práticas de engenharia de requisitos na tarefa de descrever suas necessidades.

d) o gerenciamento de requisitos corresponde ao conjunto de atividades que auxilia a equipe do projeto a identificar, controlar e rastrear os requisitos, bem como a fazer as alterações nos requisitos durante o projeto.

e) o gerenciamento de requisitos implica a alteração, inclusão e/ou exclusão de requisitos ao produto de software, o que pode levar a alterações de prazos, de recursos humanos, de equipamentos e de tecnologia.

**74. (CESGRANRIO / Petrobras – 2005)** Uma classificação frequentemente utilizada para os requisitos de um sistema os classifica em funcionais, não funcionais e de domínio. Segundo a definição que os descreve corretamente, requisitos não-funcionais são:

a) ferramentas de apoio na elaboração e acompanhamento do cronograma do projeto, como PERT e CPM, e que permitem avaliar o processo de desenvolvimento

b) requisitos de teste utilizados para detectar problemas de lógica de programação e de modelagem de funcionalidades nas fases iniciais do processo de desenvolvimento.

c) requisitos que definem o que o sistema deve fazer e como deve se comportar diante de determinadas entradas fornecidas pelos usuários.

d) requisitos derivados do domínio da aplicação do sistema que refletem as características e as restrições deste domínio.

e) requisitos que não estão diretamente relacionados com as funções específicas do sistema.

**75. (ESAF / AFRFB – 2005)** Durante a análise de requisitos, são especificados a função e o desempenho do software, bem como a sua interface com outros elementos do sistema. Nessa etapa, também, são estabelecidas as restrições de projeto, a que o software deve atender.

**76. (ESAF / AFRFB – 2005 – Letra E)** Durante a especificação dos requisitos, são estabelecidos os critérios que permitirão ao desenvolvedor e ao cliente avaliar a qualidade, assim que o software for construído.





## GABARITO

1. LETRA E
2. LETRA B
3. LETRA D
4. LETRA A
5. LETRA A
6. LETRA B
7. LETRA E
8. LETRA B
9. LETRA D
10. LETRA E
11. LETRA B
12. LETRA D
13. LETRA A
14. LETRA A
15. LETRA B
16. LETRA E
17. LETRA B
18. LETRA D
19. LETRA E
20. LETRA B
21. LETRA B
22. LETRA E
23. LETRA D
24. LETRA D
25. LETRA B
26. LETRA D
27. LETRA A
28. LETRA E
29. LETRA B
30. LETRA A
31. LETRA D
32. LETRA A
33. LETRA A
34. LETRA D
35. LETRA B
36. LETRA A
37. LETRA D
38. LETRA A
39. LETRA B
40. LETRA B
41. LETRA D
42. LETRA C
43. LETRA B
44. LETRA E
45. LETRA C
46. LETRA C
47. LETRA A
48. LETRA B
49. LETRA B
50. LETRA B
51. LETRA C
52. LETRA E
53. LETRA B
54. LETRA D
55. LETRA D
56. LETRA C
57. LETRA A
58. LETRA A
59. LETRA C
60. LETRA C
61. LETRA C
62. LETRA C
63. LETRA D
64. LETRA B
65. LETRA B
66. LETRA E
67. LETRA E
68. LETRA B
69. LETRA B
70. LETRA D
71. LETRA B
72. LETRA A
73. LETRA C
74. LETRA E
75. CORRETO
76. CORRETO



## MINIMUM VIABLE PRODUCT (MVP)

### Conceitos Básicos

INCIDÊNCIA EM PROVA: BAIXA

**Meus queridos alunos, vocês concordam comigo que é natural que patrocinadores de projetos queiram lançar um produto totalmente completo para seus clientes. É o que os clientes esperam, não é?** Esta é a pergunta errada a ser feita! A pergunta certa seria: *qual é o mínimo que meus clientes mais valiosos precisam primeiro?* Pessoal, raramente eles precisarão de tudo! Exemplo: nesse momento, eu estou escrevendo essa aula no Microsoft Word 2019.

Em 2006, a equipe de experiência de usuário do Microsoft Office publicou resultados de uma pesquisa que descobriu que apenas cinco comandos representavam 32% de todo o uso do Microsoft Word (Copiar, Salvar, Colar, Desfazer e Negritar). *Dá para acreditar nisso?* **Esse software tem uma infinidade absurda de comandos, mas 1/3 do uso dos clientes trata apenas de cinco comandos bastante simples.**

O Minimum Viable Product (MVP) é um conceito (inicialmente cunhado e definido por Frank Robinson no contexto de desenvolvimento de software) **que ajuda a identificar e alcançar a atividade mínima viável** que inicialmente pode provar e, posteriormente, entregar os benefícios necessários para alcançar uma solução viável. *Difícil de entender?* **Basicamente, MVP não é uma versão mais barata ou diluída do produto final.**

Trata-se de uma representação dele – no todo ou em parte – que oferece valor ou representa a atividade tangível ou solução de alguma forma. Um grande autor chamado Eric Ries o define como **a versão de um novo produto que permite que uma equipe colete o máximo de aprendizado sobre os clientes com a menor quantidade de esforço.** O objetivo dessa abordagem é construir rapidamente um software que apresente seus aspectos mais importantes.

**O MVP poderá ajudá-lo a obter um feedback inicial e, em seguida, corrigir – conforme a necessidade – antes de construir um produto mais pesado.** Ele pode ser um serviço completo que atende a grupos de usuários limitados ou serviços parciais que abordam grupos de usuários mais amplos. O feedback dos clientes é extremamente importante, portanto é interessante lançar o MVP para usuários reais.

**Se você se concentrar no mínimo produto viável, significa que você dará o mínimo nas mãos dos clientes e começará a, de fato, aprender com seu comportamento real dos usuários.** Aquele cara lá de cima chamado Eric Ries conta uma história muito interessante sobre um não-sucesso. Ele conta que trabalhou por cinco anos em uma startup, com uma equipe escolhida à mão e tudo do bom e do melhor.

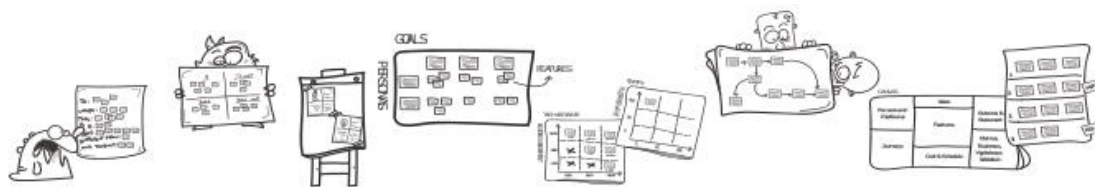


**Quando ele se juntou à equipe, eles não lhe disseram o que era o produto, apenas os nomes das pessoas legais com quem ele trabalharia. A equipe ficava em um hangar aeronaves.** O projeto era todo secreto: eles estavam construindo um produto chamado Big Idea – o futuro da Internet. Eles geraram entusiasmo no Vale do Silício, a imprensa ficava curiosa e quando o produto finalmente foi lançado, houve linhas em linhas em jornais e revistas.

No entanto, o que produto não obteve foi o cliente. Cinco anos para construir o Big Idea com todos os recursos, industrialmente fortalecidos e depois simplesmente falhar. *Não há muita sabedoria nisso tudo, vocês concordam comigo?* Esta é uma lição repetida de novo e de novo e de novo. As organizações continuam envolvendo o desenvolvimento de seus produtos em segredo, esperando até que eles tenham o conjunto completo de recursos antes de bater no mercado.

Galera, um projeto ágil é conduzido através de entregas rápidas e frequentes. Um projeto bem executado coloca ênfase em entregas de valor de acordo com objetivos de negócios e os usuários de destaque. **O MVP – versão mais simples de um produto que pode ser disponibilizada para o negócio – promove essa liberação incremental de software.** É diferente de produtos criados da forma tradicional, tipicamente com um período longo de criação de protótipo, análise e elaboração.

Como já vimos, ele foca no produto mínimo, validação de pedaços menores – bem menos elaborados do que uma versão final –, mas viável para verificar se o direcionamento está correto.



## LEAN INCEPTION

Com base no entendimento comum dos objetivos do negócio e dos usuários de destaque, busca-se o esclarecimento dos pedaços menores que compõem o produto. Por fim, vamos falar rapidamente um pouquinho sobre o Lean Inception! *O que diabos é isso, Diego?* Galera, não existe ninguém melhor para explicar esse conceito do que o próprio criador dele. E adivinhem só: ele é brasileiro e se chama Paulo Caroli:

*Projetos ágeis enfatizam a entrega precoce e contínua de software valioso, cujo valor vem dos objetivos de negócios e das necessidades dos clientes. A criação do produto Lean StartUp ajuda nessa direção, promovendo a liberação incremental de um MVP (Minimum Viable Product) – uma versão simples de um produto que é dada aos usuários para validar as principais premissas do negócio.*

*Mas como nós resolvemos o que deve estar em MVP e começamos um projeto ágil o mais rapidamente possível? Como garantir que a equipe comece a criar o produto com um entendimento compartilhado e um plano eficaz? Eu projetei o Lean Inception para responder a essas perguntas.*

*O ágil simples não tem um trabalho inicial, mas na prática percebemos que temos de fazer um pouco disso. Para a ThoughtWorks, Inception é esse "pouco disso". Desde que ingressei na ThoughtWorks, percebi que todos os projetos*



ágeis da empresa começaram de forma semelhante. A equipe do projeto se reuniria por algumas semanas, passando por muitas atividades antes de iniciar o trabalho de entrega: isso foi a Inception. Nossas Inceptions variam de projeto para projeto, mas elas geralmente geram alinhamento entre o negócio e as pessoas técnicas, e criam uma lista ordenada de estórias de usuários com estimativas juntamente com um plano de lançamento.

Eu fiquei muito satisfeito facilitando essas Inceptions ágeis dessa forma até 2011 – ano em que meu filho nasceu. A questão é que eu era o facilitador da Inception e ela levava de duas a quatro semanas. E eu não poderia ficar longe de casa por mais de uma semana. Eu tive que fazer as Inceptions mais enxutas, de alguma forma fazê-las caber em uma semana. Eu estava indo fazer a minha primeira viagem após meu filho ter nascido. Em um longo voo de São Paulo a São Francisco, li o livro *Lean StartUp*, de Eric Ries. A partir disso, eu encontrei a desculpa perfeita para reduzir o comprimento do Inception e voltar para casa depois de uma semana.

### **Por que é chamado de Lean Inception?**

Esse novo estilo de inception é definitivamente uma mudança em relação a Inception de antigamente. A equipe não mais escreve e estima histórias de usuários. Ao experimentar esse novo estilo, o nome "Inception" deu a todos a mensagem errada. Eu precisava de um nome diferente. O novo estilo de Inception é Lean (leve) por duas razões:

- A duração da Inception é menor, removendo tudo o que não se relacionava ao produto (como arquitetura, projeto etc.), tornando-se Lean;

- O resultado final da Inception é a compreensão do MVP, um conceito fundamental do movimento Lean StartUp. Portanto, o novo estilo de Inception tinha um nome claro: *The Lean Inception*.

### **Por que um Lean Inception?**

Uma Lean Inception é útil quando a equipe precisa desenvolver iterativamente um MVP. Embora o termo seja muitas vezes mal compreendido, a propriedade central de um MVP é que é algo que construímos para saber se vale a pena continuar construindo um produto. Portanto, escolhemos recursos baseados em testar nossas premissas sobre o que é valioso para nossos usuários. Para isso, precisamos entender quem são nossos usuários, que atividade eles fazem que o produto suporta e como medir se eles acharem o produto útil. Achamos que o workshop é valioso em duas circunstâncias principais.

- Grandes projetos acham que uma Lean Inception é valiosa para começar rapidamente e para ser orientada a trabalhar em um estilo Lean. Esse começo constrói iterações iniciais projetadas para descobrir e testar quais recursos são verdadeiramente valorizados pelos seus usuários.

- Organizações menores (como as Startups) usam Lean Inceptions para pegar uma ideia que já foi testada por alguns MVPs pré-software e evoluí-lo em um produto de software.

Esse workshop é especificamente sobre a compreensão de um MVP. Ele não substitui sessões de ideação, pesquisa de clientes, revisão de arquitetura ou análise competitiva. É uma técnica específica que faz parte da compreensão do que é preciso para construir um produto de sucesso. Como ele se encaixa exatamente com essas outras atividades depende muito do contexto específico da sua organização e do esforço de desenvolvimento que você está empregando.

A Transformação Digital (TD) é um tema cada vez mais relevante, impulsionando organizações a revisarem suas crenças, estratégias de negócio, produtos, práticas de gestão e seu papel na sociedade. Este movimento é notadamente importante no setor público, onde a necessidade de



inovação digital é crescente. **Uma ferramenta chave neste processo é a Lean Inception, que se destaca por sua agilidade e eficácia em definir e construir o Produto Mínimo Viável (MVP).**

**A Lean Inception é um workshop intensivo, geralmente realizado em uma semana, focado em atividades de alinhamento e definição de objetivos.** O objetivo é desenvolver um plano claro para o projeto, marcado por entregas sequenciais que visam validar o MVP, assegurando que o produto final atenda às necessidades dos usuários. Paulo Caroli narra sua experiência transformadora ao adaptar o processo de Inception para torná-lo mais compacto e focado no MVP.

Esta adaptação foi inspirada após Caroli se tornar pai em 2011, o que o motivou a buscar métodos que exigissem menor tempo de ausência de casa. Caroli vislumbrou a possibilidade de otimizar a duração das Inceptions. **Essa nova abordagem de Inception, mais enxuta e eficiente, integra princípios do Design Thinking e Lean Startup, focando nas necessidades dos usuários através de um processo de design iterativo.**

Isso permite a realização de Inceptions mais ágeis, voltadas diretamente para o desenvolvimento progressivo do produto. O ciclo "construir – aprender – construir – aprender", baseado no conceito de MVP, é fundamental. A interação inicial com o usuário proporciona aprendizados valiosos, que são então utilizados para refinar ou expandir o produto. **Assim, cada etapa de desenvolvimento é informada pelas respostas e necessidades dos usuários, permitindo ajustes precisos e eficazes.**

**(PETROBRÁS – 2019)** Do ponto de vista de design e de usabilidade, o produto mínimo viável dispensa padrões de qualidade comercial, sendo suficiente que a funcionalidade parcial proposta tenha um nível básico de confiabilidade.

**Comentários:** O MVP (Minimum Viable Product) é uma versão simples de um produto que é dada aos usuários para validar as principais premissas do negócio. Apesar disso, no MVP os padrões de qualidade não são dispensados, eles devem sempre existir (Errado).

**(BNB – 2022)** MVP (Mínimo Produto Viável) é um protótipo do sistema que é simples o suficiente para testar as características técnicas de um produto bem como é o conceito de uma solução utilizável e valiosa para o negócio.

**Comentários:** MVP (Minimum Viable Product) não é um protótipo! Trata-se de um produto que possui apenas os principais recursos necessários para torná-lo funcional e utilizável. Geralmente é a primeira versão de um produto e é usado para testar o conceito e obter feedback dos usuários. Um protótipo, por outro lado, é uma versão básica de um produto ou serviço usado para testar ideias, processos e experiência do usuário (Errado).



## QUESTÕES COMENTADAS

1. (CESPE / SERPRO – 2023) O produto mínimo viável (MVP) é uma ferramenta fundamental no processo de validação de uma ideia de negócio, por meio da coleta de problemas, oportunidades e feedbacks.

### Comentários:

O Produto Mínimo Viável (MVP) é uma versão inicial de um produto com funcionalidades básicas suficientes para atender os primeiros usuários e coletar feedback. Essa abordagem permite validar uma ideia de negócio de forma rápida e com menor custo, identificando problemas, oportunidades e áreas de melhoria antes de investir em um desenvolvimento mais completo. O MVP é uma ferramenta fundamental para testar hipóteses de mercado e ajustar o produto de acordo com as necessidades reais dos usuários.

**Gabarito:** Correto

2. (CESPE / AGER-MT – 2023) Assinale a opção que indica uma prática ágil adotada para auxílio na validação de um modelo de negócio por meio de uma versão funcional reduzida do produto.

- a) protótipo
- b) mockup
- c) MVP
- d) unidade de teste
- e) wireframe.

### Comentários:

(a) Errado. Um protótipo é uma versão preliminar de um produto ou sistema utilizado para testar conceitos e funcionalidades, mas não necessariamente foca na validação de um modelo de negócio;

(b) Errado. Ele é uma representação visual estática do produto, geralmente usada para demonstrar a interface do usuário, mas não é funcional e não é utilizado para validar um modelo de negócio;

(c) Correto. O MVP (Minimum Viable Product) é uma versão funcional reduzida do produto que permite validar um modelo de negócio com o mínimo de esforço e recursos;

(d) Errado. Uma unidade de teste é um método para validar o comportamento de pequenas partes do código, mas não é usada para validar um modelo de negócio;

(e) Errado. Trata-se de uma representação esquelética da interface do usuário, utilizado para definir a estrutura e o layout do produto, mas não é funcional e não serve para validar um modelo.





**Gabarito:** Letra C

3. (CESPE / INPI – 2024) No processo de desenvolvimento utilizando um produto mínimo viável (MVP), entre as etapas do ciclo build, measure e learn (construir, medir e aprender), a etapa build (construir) é a mais aderente ao objetivo central do MVP.

**Comentários:**

Errado. No processo de desenvolvimento utilizando um Produto Mínimo Viável (MVP), o objetivo central é aprender rapidamente sobre o mercado e as necessidades dos clientes com o mínimo esforço e investimento. Portanto, a etapa mais aderente ao objetivo central do MVP é a etapa "learn" (aprender). A construção (build) é importante, mas serve principalmente como um meio para obter feedback real dos usuários que será medido e analisado na etapa "measure" (medir), culminando no aprendizado que orientará as próximas iterações e melhorias do produto.

**Gabarito:** Errado

4. (CESGRANRIO / TRANSPETRO – 2023) Em qualquer negócio, é necessário compreender as necessidades dos usuários de forma a atendê-los adequadamente. Nesse sentido, o conceito de Produto Mínimo Viável (MVP) passou a ser muito utilizado no desenvolvimento de produtos de softwares, sendo que, nessa área, o MVP é:

- a) a versão mais simples de um produto, que será criada e disponibilizada aos usuários para validar uma ideia e coletar dados imprescindíveis para validar o direcionamento do negócio.
- b) muito utilizado em startups que desejam criar um novo produto, sendo, porém, inadequado para empresas já estabelecidas testarem novas ideias, pois sua plataforma de usuários já fornece as informações necessárias para o lançamento de novos produtos.
- c) o primeiro lançamento de um novo produto de software destinado ao grande público, correspondendo à V1.0 desse software lançado no mercado, e os esforços destinados à sua construção devem ser compatíveis para garantir alta manutenibilidade.
- d) uma versão necessariamente reduzida de um produto de software, implementado com a utilização das mesmas tecnologias que serão usadas no produto final.
- e) a versão de um produto de software lançado no mercado após uma validação de hipóteses, realizada na etapa de pesquisas, durante o processo de ideação desse produto.

**Comentários:**





- (a) Correto. O Produto Mínimo Viável (MVP) é a versão mais simples de um produto, criada e disponibilizada aos usuários para validar uma ideia e coletar dados imprescindíveis para validar o direcionamento do negócio;
- (b) Errado. O MVP é utilizado tanto por startups quanto por empresas estabelecidas para testar novas ideias e validar hipóteses, independente da plataforma de usuários existente;
- (c) Errado. O MVP não é necessariamente a primeira versão completa (V1.0) destinada ao grande público, mas uma versão básica para validação inicial de hipóteses;
- (d) Errado. O MVP é uma versão reduzida do produto, mas não é necessariamente implementado com as mesmas tecnologias do produto final; o foco é na validação da ideia;
- (e) Errado. O MVP é lançado no mercado para validar hipóteses, mas é uma etapa anterior à validação completa do produto no mercado, não após essa validação.

**Gabarito:** Letra A

5. (CESPE / EMPREL – 2023) Uma equipe de desenvolvimento de software está criando um aplicativo para conceder financiamento de imóveis para residentes no Brasil que não conseguem comprovar renda mensal regular. Nessa situação hipotética, para empregar conceitos de MVP (minimum viable product), essa equipe deve entregar uma versão do produto que:
- a) permita que um número limitado de moradores de uma comunidade consiga simular e, eventualmente, contratar o valor de um financiamento imobiliário, receber o valor correspondente (de acordo com algumas opções de garantia de pagamento já aceitas) e iniciar o recebimento do pagamento do financiamento.
- b) permita que um número limitado de moradores de uma comunidade consiga simular o valor de um financiamento imobiliário, faltando apenas a concessão dos valores finais do financiamento e a cobrança correspondente.
- c) permita que os moradores de uma comunidade aprendam sobre as opções tradicionais de financiamento imobiliário e como aumentar a sua renda familiar, devendo ser selecionado um percentual dos moradores que concluírem o aprendizado para simular o valor de um financiamento imobiliário, receber o valor correspondente e iniciar o recebimento do pagamento do financiamento.
- d) ofereça um novo modelo de financiamento para os profissionais do ramo imobiliário que possa contribuir com o novo sistema, de modo que os desenvolvedores possam confirmar quais são os recursos desejados pelo público-alvo, antes de decidir lançar o produto no mercado.



e) permita que um número limitado de moradores de uma comunidade consiga simular e, eventualmente, contratar o valor de um financiamento de um bem de consumo de baixo valor, receber o valor correspondente (de acordo com algumas opções de garantia de pagamento já aceitas) e iniciar o recebimento do pagamento do financiamento.

### Comentários:

(a) Correto. Esta alternativa descreve um MVP que permite testar o fluxo completo, desde a simulação até o recebimento do financiamento, o que é útil para validar a viabilidade do processo para o público-alvo;

(b) Errado. A ausência das etapas finais de concessão e cobrança pode não fornecer informações suficientes sobre a viabilidade do processo completo do financiamento imobiliário;

(c) Errado. A inclusão de componentes educacionais desvia o foco do MVP, que deve testar apenas as funcionalidades essenciais relacionadas ao financiamento;

(d) Errado. Focar em profissionais do ramo imobiliário desvia do público-alvo principal do MVP, que são os residentes sem comprovação de renda regular;

(e) Errado. A alternativa menciona financiamento para bens de consumo de baixo valor, que não é o foco do aplicativo de financiamento imobiliário para residentes.

**Gabarito:** Letra A



## LISTA DE QUESTÕES

- (CESPE / SERPRO – 2023)** O produto mínimo viável (MVP) é uma ferramenta fundamental no processo de validação de uma ideia de negócio, por meio da coleta de problemas, oportunidades e feedbacks.
- (CESPE / AGER-MT – 2023)** Assinale a opção que indica uma prática ágil adotada para auxílio na validação de um modelo de negócio por meio de uma versão funcional reduzida do produto.
  - protótipo
  - mockup
  - MVP
  - unidade de teste
  - wireframe.
- (CESPE / INPI – 2024)** No processo de desenvolvimento utilizando um produto mínimo viável (MVP), entre as etapas do ciclo build, measure e learn (construir, medir e aprender), a etapa build (construir) é a mais aderente ao objetivo central do MVP.
- (CESGRANRIO / TRANSPETRO – 2023)** Em qualquer negócio, é necessário compreender as necessidades dos usuários de forma a atendê-los adequadamente. Nesse sentido, o conceito de Produto Mínimo Viável (MVP) passou a ser muito utilizado no desenvolvimento de produtos de softwares, sendo que, nessa área, o MVP é:
  - a versão mais simples de um produto, que será criada e disponibilizada aos usuários para validar uma ideia e coletar dados imprescindíveis para validar o direcionamento do negócio.
  - muito utilizado em startups que desejam criar um novo produto, sendo, porém, inadequado para empresas já estabelecidas testarem novas ideias, pois sua plataforma de usuários já fornece as informações necessárias para o lançamento de novos produtos.
  - o primeiro lançamento de um novo produto de software destinado ao grande público, correspondendo à V1.0 desse software lançado no mercado, e os esforços destinados à sua construção devem ser compatíveis para garantir alta manutenibilidade.
  - uma versão necessariamente reduzida de um produto de software, implementado com a utilização das mesmas tecnologias que serão usadas no produto final.
  - a versão de um produto de software lançado no mercado após uma validação de hipóteses, realizada na etapa de pesquisas, durante o processo de ideação desse produto.
- (CESPE / EMPREL – 2023)** Uma equipe de desenvolvimento de software está criando um aplicativo para conceder financiamento de imóveis para residentes no Brasil que não



conseguem comprovar renda mensal regular. Nessa situação hipotética, para empregar conceitos de MVP (minimum viable product), essa equipe deve entregar uma versão do produto que:

- a) permita que um número limitado de moradores de uma comunidade consiga simular e, eventualmente, contratar o valor de um financiamento imobiliário, receber o valor correspondente (de acordo com algumas opções de garantia de pagamento já aceitas) e iniciar o recebimento do pagamento do financiamento.
- b) permita que um número limitado de moradores de uma comunidade consiga simular o valor de um financiamento imobiliário, faltando apenas a concessão dos valores finais do financiamento e a cobrança correspondente.
- c) permita que os moradores de uma comunidade aprendam sobre as opções tradicionais de financiamento imobiliário e como aumentar a sua renda familiar, devendo ser selecionado um percentual dos moradores que concluírem o aprendizado para simular o valor de um financiamento imobiliário, receber o valor correspondente e iniciar o recebimento do pagamento do financiamento.
- d) ofereça um novo modelo de financiamento para os profissionais do ramo imobiliário que possa contribuir com o novo sistema, de modo que os desenvolvedores possam confirmar quais são os recursos desejados pelo público-alvo, antes de decidir lançar o produto no mercado.
- e) permita que um número limitado de moradores de uma comunidade consiga simular e, eventualmente, contratar o valor de um financiamento de um bem de consumo de baixo valor, receber o valor correspondente (de acordo com algumas opções de garantia de pagamento já aceitas) e iniciar o recebimento do pagamento do financiamento.



## GABARITO

1. CORRETO
2. LETRA C
3. ERRADO
4. LETRA A
5. LETRA A



## DÉBITO TÉCNICO

### Conceitos Básicos

INCIDÊNCIA EM PROVA: BAIXA

Imagine que você seja um cara bastante procrastinador (muito comum entre concurseiros) e tenha uma tarefa para fazer, mas decide deixá-la para depois porque está ocupado com outras coisas. **Esse adiamento pode gerar uma dívida para você, que terá que lidar com a tarefa mais tarde e possivelmente em um momento menos conveniente.** De maneira semelhante, o débito técnico no desenvolvimento de software é como uma dívida do desenvolvedor.

**O desenvolvedor assume ao deixar de realizar tarefas importantes, como refatorar o código, realizar testes adequados ou atualizar bibliotecas.** Essas omissões podem levar a problemas futuros, como bugs, mau desempenho ou problemas de segurança. Assim como uma dívida financeira, o débito técnico precisa ser pago mais tarde, muitas vezes com juros, na forma de correções de emergência, retrabalho ou atrasos no cronograma.

**É importante que os desenvolvedores e equipes de software gerenciem e controlem o débito técnico para garantir que o software seja robusto, de alta qualidade e fácil de manter.** Isso envolve identificar as áreas de dívida técnica, priorizar o pagamento de débitos e investir em práticas de desenvolvimento saudáveis para minimizar a criação de novas dívidas. No meu trabalho, nós mantemos um backlog dos principais débitos técnicos para ir “pagando” sempre que possível.

**Enfim, o débito técnico é basicamente o custo a longo prazo de desenvolver um software com baixa qualidade.** Quando você escreve um código de má qualidade ou deixa de seguir boas práticas de programação, você acumula um débito técnico. Mais cedo ou mais tarde, você precisará corrigir esses problemas para manter o software funcionando corretamente ou adicionar novas funcionalidades.

As consequências do débito técnico são muitas e variadas. **Quando você acumula débito técnico, o software se torna mais difícil de entender, de modificar e de manter.** Isso pode levar a problemas de desempenho, segurança e escalabilidade, além de dificultar a adição de novas funcionalidades no futuro. Logo, é importante destacar a relação intrínseca entre débito técnico no desenvolvimento de software e refatoração de software.

**Além disso, acumular débito técnico pode ser muito caro a longo prazo.** A correção de problemas de qualidade geralmente é demorada e cara, o que pode afetar negativamente o orçamento e o cronograma do projeto. Por isso, é importante gerenciar o débito técnico de forma eficaz. Identificar, priorizar e corrigir o débito técnico acumulado pode ajudar a melhorar a qualidade do software, a reduzir custos e a garantir que o software esteja preparado para as demandas futuras.

Identificar o débito técnico pode ser um pouco complicado, mas existem algumas formas de se fazer isso. Geralmente, o débito técnico pode ser identificado a partir de sintomas comuns, como



atrasos no projeto, problemas de qualidade, bugs frequentes, entre outros. **Além disso, uma das melhores maneiras de identificar o débito técnico é realizar revisões regulares do código, seja por meio de revisões de código (code review) ou testes de software.**

**Durante essas revisões, é possível encontrar problemas comuns de qualidade de software, tais como código duplicado, falta de documentação, uso inadequado de variáveis e funções, entre outros.** Quanto às categorias de débito técnico, existem várias maneiras de categorizar o débito técnico, mas uma das mais comuns é a divisão entre débito técnico "intencional" e "não intencional".

O débito técnico intencional ocorre quando você toma uma decisão consciente de priorizar a entrega rápida do software em detrimento da qualidade. Por outro lado, o débito técnico não intencional ocorre quando você não percebe que está acumulando débito técnico, geralmente por falta de conhecimento ou tempo limitado. **Também é importante medir o débito técnico para ter uma ideia clara da quantidade de trabalho necessário para corrigi-lo.**

Existem várias ferramentas que podem ajudar a medir o débito técnico, como o SonarQube e o CodeClimate, que analisam o código-fonte em busca de problemas comuns e geram relatórios sobre o débito técnico acumulado. **A métrica mais comum para medir o débito técnico é o tempo necessário para corrigir os problemas encontrados, geralmente medido em dias ou semanas.** Gerenciar o débito técnico também é uma parte essencial do desenvolvimento de software.

Para priorizar o tratamento do débito técnico, você deve primeiro avaliar os riscos associados a cada problema identificado. **Problemas mais críticos, como problemas de segurança ou problemas que afetam o desempenho do software, devem ser tratados com mais urgência do que problemas menos críticos.** Outra forma de priorizar o tratamento do débito técnico é avaliar o impacto que cada problema tem no projeto como um todo.

Problemas que afetam mais áreas do software ou que afetam a capacidade de adicionar novas funcionalidades devem ter uma prioridade maior do que problemas que afetam apenas uma parte específica do software. Para planejar e executar a correção do débito técnico, é importante ter um processo claro e bem definido. **Isso envolve identificar os problemas, avaliar a prioridade de cada problema e planejar as etapas necessárias para corrigi-los.**

**Uma das melhores práticas para gerenciar o débito técnico é estabelecer um processo contínuo de revisão e correção.** Isso significa que você deve realizar revisões regulares do código, identificar os problemas e corrigi-los o mais rápido possível. Além disso, é importante envolver toda a equipe no processo de gerenciamento do débito técnico, para garantir que todos estejam cientes dos problemas e trabalhem juntos para resolvê-los.

Outra prática importante é manter um registro de todos os problemas de débito técnico identificados e corrigidos. Isso pode ajudar a identificar padrões e tendências ao longo do tempo e a avaliar a eficácia do processo de gerenciamento do débito técnico. **Por fim, é importante ter em**





**mente que o gerenciamento do débito técnico é um processo contínuo e nunca termina: sempre haverá novos problemas e desafios a serem enfrentados.**

No entanto, com um processo claro e uma equipe comprometida, é possível gerenciar o débito técnico de forma eficaz e garantir a qualidade e a escalabilidade do software ao longo do tempo. **Prevenir a acumulação de débito técnico é a melhor forma de garantir que seu software seja eficiente, escalável e fácil de manter a longo prazo.** Aqui estão algumas estratégias para evitar o acúmulo de débito técnico:

ESTRATÉGIAS	DESCRIÇÃO
<b>ESTABELEÇA PADRÕES DE CÓDIGO E BOAS PRÁTICAS DE DESENVOLVIMENTO</b>	Ter padrões de código claros e boas práticas de desenvolvimento estabelecidos desde o início é uma ótima maneira de garantir que o código seja claro, fácil de entender e fácil de manter.
<b>FAÇA REVISÕES DE CÓDIGO REGULARES</b>	Revisões de código regulares ajudam a identificar problemas de qualidade do código antes que eles se tornem problemas maiores. Além disso, elas promovem a colaboração da equipe e ajudam a garantir que todos estejam trabalhando dentro dos mesmos padrões e práticas.
<b>INVISTA EM TESTES DE QUALIDADE</b>	Testes automatizados e manuais são uma ótima maneira de garantir que o software esteja funcionando corretamente e atendendo às necessidades dos usuários.
<b>PLANEJE PARA ESCALABILIDADE</b>	Ao desenvolver um software, é importante pensar em como ele pode ser escalável no futuro. Isso significa pensar em como o software pode lidar com grandes quantidades de dados e usuários, bem como garantir que ele possa ser facilmente mantido e atualizado.
<b>USE FERRAMENTAS DE AUTOMAÇÃO</b>	Ferramentas de automação podem ajudar a reduzir o tempo necessário para realizar tarefas repetitivas e propensas a erros. Isso pode incluir ferramentas de integração contínua, ferramentas de análise de código e ferramentas de teste automatizado.
<b>MANTENHA UM REGISTRO DE DÍVIDA TÉCNICA</b>	Manter um registro de dívida técnica pode ajudar a garantir que os problemas sejam identificados e tratados o mais rápido possível.

**(Câmara de Goiânia/GO – 2018)** Leia o texto a seguir extraído da Internet.

*Se o débito técnico não é pago, ele pode acumular, tornando mais difícil implementar mudanças posteriores.*

No contexto desta informação, o débito técnico:

- a) pode ser eliminado durante a construção de software sem dependência do projeto (design) do software.
- b) pode ser eliminado por alteração no projeto (design) sem repercussão na implementação.



- c) é uma questão de projeto (design) com repercussão na funcionalidade do software.
- e) tem impacto na evolução do software.

**Comentários:** no contexto desta informação, o débito técnico tem impacto na evolução do software. Quando o débito técnico não é tratado, ele se acumula e pode tornar mais difícil a implementação de mudanças posteriores. Isso pode afetar negativamente a evolução do software, levando a atrasos no lançamento, custos adicionais e, possivelmente, um produto final de menor qualidade. Logo, é importante gerenciar o débito técnico de maneira eficaz para garantir que o software possa evoluir de maneira consistente e sem problemas (Letra D).

**(TJ/PE – 2012)** No contexto de programação ágil XP, um débito técnico é descrito como o:

- a) número de pontos funcionais não entregues no último período.
- b) custo homem/hora da equipe técnica para um determinado projeto.
- c) método de modificação do código fonte, com alteração do seu comportamento, porém sem alteração de seu significado.
- d) dispêndio relacionado ao desenvolvimento, teste ou entrega da parte funcional do sistema.
- e) total de desenvolvimento feito de maneira rápida e simples sem, às vezes, levar em consideração testes e arquitetura do sistema.

**Comentários:** o débito técnico em XP é entendido como uma dívida técnica que é adquirida quando o desenvolvimento é realizado de forma rápida e simplificada, muitas vezes sem levar em consideração os testes e a arquitetura adequados do sistema. Essa dívida técnica pode se acumular e, se não for tratada, pode levar a problemas futuros no projeto. Logo, é importante gerenciar o débito técnico para garantir que ele não se acumule a ponto de afetar negativamente o projeto (Letra E).

**(CEBRASPE / SERPRO – 2023)** A dívida técnica pode ser vista como um empréstimo que o time de desenvolvimento faz para si mesmo com o objetivo de acelerar o processo de desenvolvimento, porém, por ser considerada uma estratégia ruim, deve ser evitada, devido a suas possíveis consequências negativas.

**Comentários:** a dívida técnica é, de fato, comparada a um empréstimo que o time de desenvolvimento faz para acelerar a entrega, consciente de que terá que "pagar" esse débito mais tarde com esforços adicionais de refatoração ou correções. No entanto, ela não é necessariamente uma estratégia ruim. Pode ser uma escolha deliberada e útil em determinadas situações, como quando há uma necessidade urgente de entrega. Por conta da péssima redação, o item foi anulado sob a seguinte justificativa: "A ausência de informações relevantes na redação do item prejudicou seu julgamento objetivo" (Anulado).





## TÉCNICAS DE PRIORIZAÇÃO DE BACKLOG

### Conceitos Básicos

INCIDÊNCIA EM PROVA: BAIXA

A **priorização do backlog do produto é uma atividade crítica no gerenciamento ágil de projetos, ajudando a garantir que a equipe se concentre nas tarefas mais importantes que agregam valor ao produto final.** Galera, quem já trabalhou com projeto de software sabe como é difícil, custoso, complexo, trabalhoso, chato, árduo e ingrato fazer com que o cliente consiga priorizar suas necessidades. Em geral, clientes acreditam que absolutamente tudo é importante!

### MoSCoW

INCIDÊNCIA EM PROVA: BAIXA



O **Método MoSCoW é uma técnica de priorização que ajuda equipes de projeto a categorizar itens do backlog ou requisitos do projeto em quatro categorias distintas, baseadas em sua importância e urgência para a entrega do projeto.** Essas categorias são: Must Have (Deve Ter), Should Have (Deveria Ter), Could Have (Poderia Ter) e Won't Have (Não Terá Desta Vez). Vamos ver um determinado contexto para explicar melhor cada categoria.

### EXEMPLO CONCRETO

**Imagine que uma equipe de desenvolvimento de software esteja trabalhando em um projeto para criar um novo aplicativo de gerenciamento de projetos.** Durante a fase de planejamento do projeto, eles usam o Método MoSCoW para priorizar os recursos e funcionalidades que o aplicativo deve ter. Nos parágrafos seguintes, nós vamos ver a definição de cada uma das categorias e veremos exemplos dessas categorias no exemplo da equipe de desenvolvimento de software.



### MUST HAVE [DEVE TER]

Itens classificados como "Must Have" são essenciais para o sucesso do projeto. Sem eles, o projeto não pode ser considerado completo ou bem-sucedido. Esses itens são críticos e não negociáveis.

- **Criação e Gerenciamento de Tarefas:** capacidade de criar, atribuir e gerenciar tarefas dentro de projetos.
- **Controle de Prazos e Calendário:** ferramentas para definir prazos e visualizar tarefas em um calendário.
- **Sistema de Notificações:** notificações para alertar os usuários sobre prazos iminentes e mudanças nas tarefas.

### SHOULD HAVE [DEVERIA TER]

Itens "Should Have" são importantes, mas não críticos. São recursos ou requisitos que agregam valor significativo ao projeto, mas sua ausência não inviabiliza o lançamento do produto ou serviço.

- **Relatórios de Progresso do Projeto:** funcionalidade para gerar relatórios visuais sobre o progresso do projeto.
- **Integração com E-mails:** capacidade de enviar atualizações de tarefas e notificações diretamente por e-mail.

### COULD HAVE [PODERIA TER]

Itens classificados como "Could Have" são desejáveis, mas menos importantes e com menor impacto imediato no projeto. Geralmente, são incluídos se o tempo e os recursos permitirem.

- **Relatórios de Progresso do Projeto:** funcionalidade para gerar relatórios visuais sobre o progresso do projeto.
- **Integração com E-mails:** capacidade de enviar atualizações de tarefas e notificações diretamente por e-mail.

### WON'T HAVE [NÃO TERÁ DESTA VEZ]

Itens "Won't Have" são aqueles que foram identificados e reconhecidos como de menor prioridade para o ciclo atual do projeto. São excluídos da entrega iminente, mas podem ser reconsiderados em fases futuras.

- **Relatórios de Progresso do Projeto:** funcionalidade para gerar relatórios visuais sobre o progresso do projeto.
- **Integração com E-mails:** capacidade de enviar atualizações de tarefas e notificações diretamente por e-mail.

Ao categorizar as funcionalidades usando o Método MoSCoW, a equipe de desenvolvimento pode claramente identificar quais recursos são essenciais para o lançamento inicial do aplicativo e quais podem ser adiados para versões futuras. **Isso ajuda a equipe a focar seus esforços nos aspectos mais críticos do aplicativo, garantindo que o produto final atenda às necessidades básicas dos usuários enquanto permanece dentro do orçamento e do cronograma do projeto.**



Além disso, ajuda na comunicação com as partes interessadas, oferecendo uma visão clara do que esperar no lançamento e o que pode ser incluído em atualizações futuras.



## Scorecard

INCIDÊNCIA EM PROVA: BAIXA

A técnica de Scorecard para priorização de backlog é um método estruturado que ajuda equipes a avaliar e priorizar itens de backlog com base em um conjunto de critérios pré-definidos. Diferentemente de abordagens mais subjetivas, o Scorecard atribui valores numéricos ou pontuações a cada item do backlog segundo critérios específicos, facilitando comparações objetivas entre diferentes tarefas ou histórias de usuário.

**Definição de Critérios:** primeiramente, a equipe define critérios relevantes para a avaliação dos itens do backlog. Esses critérios podem incluir:

<b>DEFINIÇÃO DE CRITÉRIOS</b>	Primeiramente, a equipe define critérios relevantes para a avaliação dos itens do backlog. Esses critérios podem incluir: valor para o cliente; objetivos do negócio; esforço ou complexidade; risco; e dependências.
<b>ATRIBUIÇÃO DE PONTUAÇÕES</b>	Cada item de backlog é avaliado contra os critérios definidos, com pontuações atribuídas de acordo com o grau em que o item atende a cada critério. As pontuações podem ser simples (Ex: de 1 a 5) ou baseadas em uma escala mais complexa.
<b>CÁLCULO DA PONTUAÇÃO</b>	Após avaliar todos os itens contra cada critério e considerar os pesos atribuídos, calcula-se a pontuação total para cada item. Isso é feito multiplicando a pontuação de cada critério pelo seu peso correspondente e somando os resultados.
<b>RESULTADO DA PRIORIZAÇÃO</b>	Os itens do backlog são então classificados com base em suas pontuações totais, do mais alto ao mais baixo, permitindo à equipe identificar quais itens devem ser priorizados para desenvolvimento ou ação.

### EXEMPLO CONCRETO

Imaginemos uma equipe de desenvolvimento de software que está trabalhando em um novo aplicativo de finanças pessoais. Com muitas ideias para funcionalidades, a equipe precisa priorizar o backlog de produto de maneira eficaz. Eles decidem usar a técnica de Scorecard para ajudar na priorização. **Na Definição de Critérios, a equipe define os seguintes critérios, cada um com uma escala de 1 a 5, onde 5 indica a maior importância ou impacto, e 1 a menor:**

**Definição de Critérios** | A equipe define os seguintes critérios, cada um com uma escala de 1 a 5, onde 5 indica a maior importância ou impacto, e 1 a menor:

- **Valor para o Cliente/Usuário:** o quanto a funcionalidade agrega valor ou resolve um problema para o usuário final;
- **Alinhamento Estratégico:** como a funcionalidade se alinha com os objetivos de longo prazo da empresa;
- **Viabilidade Técnica:** trata-se da facilidade ou dificuldade técnica de implementar a funcionalidade;
- **Impacto no Mercado:** o potencial da funcionalidade em aumentar a participação de mercado ou atrair novos usuários;





- **Risco:** os riscos associados à implementação da funcionalidade, incluindo riscos técnicos e de negócios.

**Itens do Backlog para Priorização** | A equipe seleciona três funcionalidades para avaliação:

- **Funcionalidade A:** implementação de um recurso de orçamento automatizado;
- **Funcionalidade B:** adição de uma ferramenta de investimento com inteligência artificial;
- **Funcionalidade C:** integração de pagamentos móveis;

**Atribuição e Cálculo da Pontuação** | A equipe avalia cada funcionalidade contra os critérios definidos e faz a conta total:

CRITÉRIO/CATEGORIA	FUNCIONALIDADE A	FUNCIONALIDADE B	FUNCIONALIDADE C
VALOR PARA O CLIENTE	5	4	3
ALINHAMENTO ESTRATÉGICO	4	3	5
VIABILIDADE TÉCNICA	3	2	4
IMPACTO NO MERCADO	3	5	4
RISCO	2	4	3
TOTAL	17	18	19

**Resultado da Priorização** | Com base nas pontuações totais, a Funcionalidade C (Integração de pagamentos móveis) é a mais alta prioridade, seguida pela Funcionalidade B (ferramenta de investimento com IA) e, por último, a Funcionalidade A (recurso de orçamento automatizado).

Este exemplo mostra como a técnica de Scorecard pode ser usada para avaliar objetivamente diferentes funcionalidades contra critérios relevantes, ajudando a equipe a tomar decisões informadas sobre a priorização do backlog. **Essa abordagem garante que os itens mais valiosos e estratégicos sejam priorizados, levando em conta não apenas o valor para o cliente, mas também a viabilidade, o impacto no mercado e os riscos associados.**



## BUC

INCIDÊNCIA EM PROVA: BAIXA

O BUC (Benefício x Custo x Urgência) é uma técnica estruturada que ajuda as equipes a avaliar e classificar itens do backlog de um projeto com base em três critérios principais: **Benefício, Custo e Urgência**. Essa metodologia proporciona uma visão equilibrada sobre quais funcionalidades ou tarefas devem ser priorizadas, considerando não apenas o valor que elas trazem, mas também os recursos necessários para sua implementação e a importância do tempo.

<b>BENEFÍCIO</b>	Este critério avalia o valor ou o impacto positivo que a implementação de um item do backlog trará para o projeto, a organização, os usuários finais ou os clientes. Itens que oferecem maiores benefícios geralmente recebem uma priorização mais alta.
<b>CUSTO</b>	Refere-se aos recursos necessários para implementar um item do backlog. Isso pode incluir tempo de desenvolvimento, esforço da equipe, custos financeiros, entre outros. Itens que exigem menos recursos (custo menor) podem ser mais atraentes para implementação rápida.
<b>URGÊNCIA</b>	Este critério considera o quão crítico é implementar um item do backlog dentro de um determinado período de tempo. Itens urgentes podem ser aqueles que são necessários para cumprir prazos legais, de mercado ou internos específicos.

Vejamos as etapas do processo de priorização do BUC:

<b>AVALIAÇÃO</b>	Cada item do backlog é avaliado em relação a cada um dos três critérios. Isso geralmente é feito através de discussões em equipe, podendo-se utilizar escalas numéricas para quantificar cada critério.
<b>PONTUAÇÃO E COMPARAÇÃO</b>	Após a avaliação, cada item recebe uma pontuação geral que combina os três critérios. Há maneiras de combinar essas pontuações, incluindo fórmulas matemáticas simples ou modelos ponderados que dão mais importância a critérios específicos.
<b>PRIORIZAÇÃO</b>	Os itens contidos do backlog são – portanto – classificados com base em suas pontuações totais. Dessa forma, é possível ajudar a equipe a identificar quais devem ser abordados primeiro.

O sucesso da aplicação do BUC depende da capacidade da equipe de avaliar precisamente cada um dos critérios e de aplicar consistentemente essas avaliações em todo o backlog. Além disso, a técnica requer um equilíbrio cuidadoso entre os critérios para evitar sobrevalorizar um em detrimento dos outros, garantindo que as decisões de priorização se alinhem estrategicamente com os objetivos de longo prazo do projeto.

### EXEMPLO CONCRETO

Imaginemos uma equipe de desenvolvimento de software encarregada de criar um novo sistema de gerenciamento de Recursos Humanos (RH) para uma grande empresa. O backlog do projeto está cheio de funcionalidades sugeridas, mas a equipe precisa priorizá-las para o próximo



ciclo de desenvolvimento. Para fazer isso, eles decidem aplicar a técnica BUC. Vejamos a definição dos Critérios BUC:

<b>BENEFÍCIO [B]</b>	O impacto positivo que a funcionalidade terá na eficiência operacional do departamento de RH e na satisfação dos funcionários.
<b>URGÊNCIA [U]</b>	A necessidade de implementar a funcionalidade para atender a prazos legais ou de negócios.
<b>CUSTO [C]</b>	Os recursos necessários para desenvolver a funcionalidade, incluindo tempo, mão de obra e outros custos associados.

Dito isso, vamos considerar que temos os seguintes itens do backlog para priorização:

1. Portal do Funcionário para Autoatendimento
2. Integração com Software de Folha de Pagamento
3. Sistema de Rastreamento de Candidatos

Agora para cada item, nós vamos fazer a Avaliação BUC e Priorização:

### 1. Portal do Funcionário para Autoatendimento

<b>BENEFÍCIO</b>	<b>ALTO [5]</b>	Melhorará significativamente a satisfação dos funcionários e reduzirá a carga de trabalho administrativo do RH.
<b>URGÊNCIA</b>	<b>MÉDIO [3]</b>	Importante para a satisfação dos funcionários, mas não existe um prazo legal.
<b>CUSTO</b>	<b>MÉDIO [3]</b>	Significa que requer uma quantidade relativamente moderada de desenvolvimento.

### 2. Integração com Software de Folha de Pagamento

<b>BENEFÍCIO</b>	<b>ALTO [5]</b>	Eliminação de processos manuais e redução de erros.
<b>URGÊNCIA</b>	<b>ALTO [5]</b>	Necessário para cumprir os novos regulamentos fiscais.
<b>CUSTO</b>	<b>ALTO [4]</b>	Complexidade técnica e tempo necessário são consideráveis.

### 3. Sistema de Rastreamento de Candidatos

<b>BENEFÍCIO</b>	<b>MÉDIO [3]</b>	Melhora o processo de recrutamento, mas não afeta diretamente a maioria dos funcionários existentes.
<b>URGÊNCIA</b>	<b>BAIXO [2]</b>	Não há prazos imediatos ou regulamentações que exijam essa funcionalidade.
<b>CUSTO</b>	<b>BAIXO [2]</b>	Relativamente simples de implementar, usando soluções de mercado.



Com base na Avaliação BUC, a equipe poderá decidir priorizar as funcionalidades da seguinte forma:

- **Integração com Software de Folha de Pagamento:** devido à sua alta urgência e benefício significativo, apesar do custo alto.
- **Portal do Funcionário para Autoatendimento:** devido ao seu alto benefício e custo moderado, apesar da urgência média.
- **Sistema de Rastreamento de Candidatos:** devido ao seu benefício e urgência mais baixos, junto com um custo baixo.

Este exemplo demonstra como a Técnica BUC permite à equipe fazer escolhas estratégicas e informadas sobre a priorização do backlog, considerando uma visão equilibrada de benefício, urgência e custo para cada funcionalidade proposta. **Por meio dessa técnica, garante-se que recursos limitados sejam alocados de maneira eficaz para maximizar o valor entregue pelo projeto.**



## Testes de Suposição

INCIDÊNCIA EM PROVA: BAIXA

A técnica de Testes de Suposição para a priorização de backlog é uma abordagem que se concentra na validação de hipóteses ou suposições subjacentes aos itens do backlog, especialmente em contextos em que a incerteza é alta. Esta técnica é particularmente relevante para projetos que adotam metodologias ágeis e Lean Startup, onde o aprendizado rápido e a adaptação com base em feedback real são valorizados.

O objetivo é identificar quais funcionalidades devem ser priorizadas para teste ou desenvolvimento, com base na necessidade de validar suposições críticas para o sucesso do projeto. As etapas são:

ETAPAS	DESCRIÇÃO
IDENTIFICAÇÃO DE SUPOSIÇÕES	A equipe lista as suposições importantes que estão por trás de cada item do backlog. Essas suposições podem estar relacionadas ao valor que um recurso oferecerá aos usuários, à viabilidade técnica de uma funcionalidade ou à demanda do mercado por um produto.
AValiação DE RISCO E IMPACTO	Para cada suposição, a equipe avalia o risco associado (a probabilidade de a suposição estar errada) e o impacto potencial no projeto se a suposição for incorreta. Itens com suposições de alto risco e alto impacto são candidatos prioritários para testes.
DESIGN DE TESTES	Para as suposições identificadas como prioritárias, a equipe projeta experimentos ou testes mínimos viáveis para validar rapidamente essas suposições. Isso pode incluir protótipos, pesquisas com usuários, análises de dados de mercado ou outros métodos de feedback rápido.
EXECUÇÃO E APRENDIZADO	A equipe executa os testes projetados e coleta dados que ajudam a validar ou refutar as suposições. Com base nos resultados, ajustes são feitos no backlog e na estratégia do projeto conforme necessário.

Essa técnica ajuda a identificar e mitigar riscos no início do processo de desenvolvimento, focando na validação de suposições críticas. Ela também garante que o esforço de desenvolvimento esteja focado em funcionalidades que oferecem valor real aos usuários e ao negócio, evitando desperdício de recursos. Por fim, ela promove uma cultura de aprendizado e adaptação contínua, essencial para ambientes dinâmicos e incertos.

Por outro lado, essa técnica também requer uma cultura organizacional que valorize a experimentação e esteja confortável com a falha como uma forma de aprendizado. É importante gerenciar as expectativas das partes interessadas sobre o processo de teste e os ajustes que podem ser necessários no projeto com base nos resultados dos testes. Dito isso, vamos ver um exemplo concreto de uso dessa técnica.

### EXEMPLO CONCRETO

Imaginemos uma startup que está desenvolvendo um novo aplicativo móvel para ajudar pessoas a encontrar companheiros de corrida locais. A equipe tem várias ideias sobre funcionalidades que poderiam tornar o aplicativo valioso para seus usuários, mas antes de se



comprometerem com um desenvolvimento extenso, eles decidem usar a técnica de Testes de Suposição para validar suas ideias principais. Vejamos as suposições do projeto:

<b>SUPOSIÇÃO A</b>	Usuários estão dispostos a pagar por funcionalidades premium, como análise avançada do desempenho em corridas.
<b>SUPOSIÇÃO B</b>	Existe uma demanda significativa por um sistema de matchmaking que conecta corredores com habilidades similares e interesses em corridas.
<b>SUPOSIÇÃO C</b>	Os usuários utilizarão ativamente a funcionalidade de eventos para organizar e participar de corridas locais.

Agora fazemos uma avaliação de Risco/Impacto:

<b>SUPOSIÇÃO A</b>	<b>ALTO RISCO/ ALTO IMPACTO</b>	Se os usuários não estiverem dispostos a pagar, o modelo de receita previsto pode falhar.
<b>SUPOSIÇÃO B</b>	<b>MÉDIO RISCO/ ALTO IMPACTO</b>	Essencial para a proposta de valor do aplicativo, mas alternativas de validação são viáveis.
<b>SUPOSIÇÃO C</b>	<b>BAIXO RISCO/ MÉDIO IMPACTO</b>	Contribui para a retenção de usuários, mas não é crucial para o lançamento inicial.

Feito isso, agora vamos falar sobre Design de Testes:

<b>PARA A SUPOSIÇÃO A</b>	A equipe decide criar uma pesquisa online direcionada aos potenciais usuários do aplicativo, questionando sobre a disposição de pagar por funcionalidades premium e quais funcionalidades seriam mais valiosas.
<b>PARA A SUPOSIÇÃO B</b>	A equipe desenvolve um protótipo mínimo viável (MVP) do sistema de matchmaking e convida um pequeno grupo de usuários locais para testá-lo, observando a frequência de uso e coletando feedback direto.
<b>PARA A SUPOSIÇÃO C</b>	Decide-se postergar testes mais elaborados até que as duas suposições mais críticas sejam validadas, usando inicialmente feedback de fóruns e grupos de corrida online para medir o interesse.

Finalizado, vamos falar sobre Execução e Aprendizado:

<b>SUPOSIÇÃO A</b>	A pesquisa revela que a maioria dos usuários potenciais não está disposta a pagar por funcionalidades premium, preferindo um modelo suportado por anúncios. A equipe ajusta o modelo de receita.
<b>SUPOSIÇÃO B</b>	O teste do MVP mostra que os usuários estão muito engajados com o sistema de matchmaking, validando a demanda por essa funcionalidade. A equipe prioriza o desenvolvimento completo dessa funcionalidade.
<b>SUPOSIÇÃO C</b>	Feedback preliminar indica interesse na funcionalidade de eventos, mas com menor prioridade do que o sistema de matchmaking.

**Ao aplicar Testes de Suposição, a equipe foi capaz de validar suas ideias críticas de forma eficaz, ajustar suas estratégias com base em dados reais e priorizar o desenvolvimento de**



**funcionalidades que oferecem o maior valor para seus usuários.** Isso permitiu que a startup utilizasse seus recursos de forma mais eficiente e aumentasse as chances de sucesso do aplicativo no mercado.





## Valor de Negócio x Risco

INCIDÊNCIA EM PROVA: BAIXA

**Essa técnica de priorização de backlog é um método estratégico utilizado para classificar e priorizar itens de backlog com base em dois critérios principais: o valor que cada item traz para o negócio e o risco associado à sua implementação.** Essa abordagem ajuda as equipes a tomar decisões informadas sobre quais funcionalidades ou tarefas devem ser desenvolvidas primeiro, equilibrando potenciais benefícios com os desafios ou incertezas envolvidos. *E como funciona?*

ETAPAS	DESCRIÇÃO
<b>AVALIAÇÃO DO VALOR DE NEGÓCIO</b>	O valor de negócio é determinado considerando o impacto que a implementação de um item do backlog terá sobre os objetivos estratégicos da empresa, a satisfação do cliente, a receita, a eficiência operacional, entre outros fatores. Itens que são considerados de alto valor são aqueles que contribuem significativamente para o sucesso e crescimento do negócio.
<b>AVALIAÇÃO DO RISCO</b>	O risco envolve avaliar as incertezas e desafios associados à implementação de um item do backlog. Isso pode incluir riscos técnicos, riscos de mercado, dependências de outras tarefas ou funcionalidades, e qualquer outro fator que possa impactar a entrega ou o sucesso do item. Itens com riscos altos requerem atenção especial e podem necessitar de mais recursos ou preparação.

***E como ocorre o processo de priorização? Após avaliar cada item do backlog em termos de valor de negócio e risco, a equipe utiliza essas informações para priorizar o trabalho.*** Isso geralmente é feito de forma a maximizar o valor entregue, minimizando ao mesmo tempo os riscos para o projeto. Uma forma comum de visualizar isso é através de uma matriz 2x2, onde um eixo representa o valor de negócio e o outro o risco.

Itens de alto valor e baixo risco são priorizados, enquanto itens de baixo valor e alto risco podem ser reavaliados ou adiados. Essa técnica tem os seguintes benefícios:

- **Foco no Valor:** garante que a equipe se concentre em itens que oferecem o maior retorno sobre o investimento.
- **Gerenciamento de Riscos:** ajuda a identificar e mitigar riscos no início do ciclo de desenvolvimento.
- **Alocação Eficiente de Recursos:** direciona recursos para áreas que proporcionam o maior impacto positivo no negócio, ao mesmo tempo em que considera os desafios envolvidos.
- **Decisões Informadas:** fornece uma base sólida para tomada de decisões sobre o que incluir em sprints ou iterações futuras.

**Embora essa técnica ofereça uma abordagem equilibrada para a priorização do backlog, ela requer uma compreensão clara dos objetivos estratégicos do negócio e uma avaliação precisa tanto do valor quanto do risco.** Além disso, é importante revisar e ajustar a priorização regularmente à medida que novas informações se tornam disponíveis ou quando mudanças ocorrem no ambiente de negócios ou tecnológico.



## EXEMPLO CONCRETO

Vamos considerar uma empresa de software que está desenvolvendo uma nova plataforma de e-commerce destinada a pequenas e médias empresas (PMEs). O backlog do projeto está repleto de funcionalidades propostas, mas a equipe de desenvolvimento precisa priorizá-las para garantir que o produto final atenda às necessidades do mercado de forma eficaz, dentro do orçamento e do prazo estabelecidos. A equipe decide aplicar a técnica, começando pela definição dos critérios:

DEFINIÇÃO DOS CRITÉRIOS	DESCRIÇÃO
VALOR DE NEGÓCIO	Contribuição de cada funcionalidade para os objetivos estratégicos da empresa, como aumentar a base de clientes, melhorar a satisfação do cliente, aumentar a receita, etc.
RISCO	Considerações incluem complexidade técnica, incertezas de desenvolvimento, dependências externas, e potenciais impactos no prazo e custo.

ITENS DO BACKLOG PARA PRIORIZAÇÃO	DESCRIÇÃO
INTEGRAÇÃO C/ SISTEMAS DE PAGAMENTO POPULARES	Permitir que os usuários escolham entre múltiplos sistemas de pagamento.
SISTEMA DE RECOMENDAÇÃO DE PRODUTOS BASEADO EM IA	Recomendar produtos aos usuários com base em seu histórico de navegação e compra.
FERRAMENTA DE ANÁLISE DE DADOS DE VENDAS	Oferecer aos vendedores análises detalhadas de suas vendas e tendências do mercado.

INTEGRAÇÃO C/ SISTEMAS DE PAGAMENTO POPULARES	DESCRIÇÃO
VALOR DE NEGÓCIO	Alto: essencial para facilitar as vendas e melhorar a experiência do cliente.
RISCO	Baixo: existem muitas soluções prontas que podem ser integradas com esforço técnico moderado.

SISTEMA DE RECOMENDAÇÃO DE PRODUTOS BASEADO EM IA	DESCRIÇÃO
VALOR DE NEGÓCIO	Médio: pode aumentar as vendas por meio de compras impulsivas e melhorar a experiência do usuário.
RISCO	Alto: requer um desenvolvimento complexo e pode envolver questões de privacidade dos dados.

FERRAMENTA DE ANÁLISE DE DADOS DE VENDAS	DESCRIÇÃO
VALOR DE NEGÓCIO	Médio: útil para vendedores, mas não essencial para operações do dia a dia.



**RISCO**

Médio: desenvolvimento técnico é desafiador, mas existem bibliotecas e ferramentas que podem ser utilizadas.

Com base nesta análise, a equipe decide priorizar as funcionalidades da seguinte maneira:

PRIORIZAÇÃO BASEADA EM VALOR DE NEGÓCIO X RISCO	DESCRIÇÃO
INTEGRAÇÃO C/ SISTEMAS DE PAGAMENTO POPULARES	Devido ao seu alto valor de negócio e baixo risco, é escolhida como a prioridade mais alta para garantir uma entrada suave no mercado e uma ótima experiência do usuário desde o lançamento.
SISTEMA DE RECOMENDAÇÃO DE PRODUTOS BASEADO EM IA	Embora o valor de negócio seja considerado médio, o risco técnico moderado torna esta funcionalidade uma segunda prioridade, fornecendo valor adicional aos vendedores sem um risco excessivo.
FERRAMENTA DE ANÁLISE DE DADOS DE VENDAS	Apesar de seu potencial para aumentar as vendas, o alto risco associado a complexidades técnicas e questões de privacidade faz com que seja priorizado por último. A equipe pode optar por visitar esta funcionalidade após avaliar o feedback do usuário e considerar abordagens para mitigar os riscos.



## QUESTÕES COMENTADAS

1. (PROF. DIEGO / INÉDITA – 2024) O Método MoSCoW pode ser usado para identificar funcionalidades que são essenciais para o lançamento inicial de um produto.

### Comentários:

O Método MoSCoW é eficazmente utilizado para identificar funcionalidades que são essenciais para o lançamento inicial de um produto, categorizando-as como "Must Have". Este processo ajuda a garantir que o foco esteja nas funcionalidades críticas que precisam estar presentes para que o produto seja viável e atenda aos requisitos básicos dos usuários e do negócio.

**Gabarito:** Correto

2. (PROF. DIEGO / INÉDITA – 2024) Itens classificados como "Should Have" no Método MoSCoW são essenciais para o sucesso do projeto e não podem ser omitidos.

### Comentários:

No Método MoSCoW, itens classificados como "Should Have" são importantes para o projeto, mas não são essenciais no mesmo sentido que os itens "Must Have". Os itens "Should Have" são aqueles que deveriam ser incluídos se possível, mas podem ser omitidos se necessário, sem inviabilizar o projeto. Eles são priorizados abaixo dos "Must Have", que são críticos e essenciais para o sucesso do projeto. Portanto, embora importantes, os itens "Should Have" podem ser adiados ou omitidos se restrições de tempo, recursos ou outras prioridades mais críticas surgirem.

**Gabarito:** Errado

3. (PROF. DIEGO / INÉDITA – 2024) No Scorecard para priorização de backlog, a atribuição de pontuações é feita sem considerar critérios pré-definidos.

### Comentários:

No Scorecard para priorização de backlog, a atribuição de pontuações é feita justamente considerando critérios pré-definidos. Cada item do backlog é avaliado e pontuado com base em um conjunto de critérios estabelecidos pela equipe, que podem incluir fatores como valor de negócio, complexidade técnica, impacto no usuário, e urgência, entre outros. Essa abordagem estruturada e baseada em critérios é essencial para garantir avaliações objetivas e consistentes dos itens do backlog.

**Gabarito:** Errado



4. (PROF. DIEGO / INÉDITA – 2024) A técnica de Scorecard permite comparações objetivas entre diferentes tarefas ao atribuir valores numéricos a cada item do backlog.

**Comentários:**

A técnica de Scorecard permite comparações objetivas entre diferentes tarefas ao atribuir valores numéricos a cada item do backlog com base em critérios previamente definidos. Essa abordagem quantitativa ajuda as equipes a avaliar e priorizar itens de forma sistemática e transparente, facilitando a tomada de decisões informadas sobre quais tarefas devem ser abordadas primeiro.

**Gabarito:** Correto

---

5. (PROF. DIEGO / INÉDITA – 2024) No BUC, itens com maior urgência sempre recebem a prioridade mais alta, independentemente de seu benefício ou custo.

**Comentários:**

No método BUC (Benefício, Urgência, Custo), itens com maior urgência não recebem automaticamente a prioridade mais alta, independentemente de seu benefício ou custo. A priorização é determinada pela avaliação balanceada de todos os três critérios. Embora a urgência seja um fator importante, ela é considerada em conjunto com o benefício que o item traz para o negócio e o custo ou recursos necessários para sua implementação. Itens que oferecem um alto benefício com urgência moderada e custo razoável podem, em muitos casos, ser priorizados sobre itens urgentes, mas com benefício limitado ou custo proibitivo.

**Gabarito:** Errado

---

6. (PROF. DIEGO / INÉDITA – 2024) O critério de Benefício no BUC avalia o impacto positivo que a implementação de um item do backlog trará para o projeto.

**Comentários:**

O critério de Benefício no método BUC efetivamente avalia o impacto positivo que a implementação de um item do backlog trará para o projeto, considerando aspectos como a contribuição para os objetivos do negócio, a melhoria na experiência do usuário, o aumento na eficiência operacional, entre outros. Esse critério ajuda a identificar quais funcionalidades ou tarefas oferecem o maior valor agregado, facilitando a priorização com base no potencial de retorno sobre o investimento.

**Gabarito:** Correto

---

7. (PROF. DIEGO / INÉDITA – 2024) A técnica de Testes de Suposição foca exclusivamente na viabilidade técnica das funcionalidades, ignorando seu valor para os usuários.



### Comentários:

A técnica de Testes de Suposição não se foca exclusivamente na viabilidade técnica das funcionalidades; ela também considera seu valor para os usuários, entre outros fatores críticos. O objetivo dos Testes de Suposição é validar hipóteses abrangentes sobre o projeto, que podem incluir suposições sobre a demanda do usuário, o impacto no mercado, além da viabilidade técnica. Portanto, essa técnica busca uma compreensão holística que engloba tanto aspectos técnicos quanto o valor oferecido aos usuários.

**Gabarito:** Errado

---

**8. (PROF. DIEGO / INÉDITA – 2024)** Testes de Suposição priorizam funcionalidades para desenvolvimento com base na necessidade de validar hipóteses críticas para o sucesso do projeto.

### Comentários:

Os Testes de Suposição priorizam funcionalidades para desenvolvimento justamente com base na necessidade de validar hipóteses críticas para o sucesso do projeto. Essa abordagem permite que as equipes identifiquem e testem suposições fundamentais o mais cedo possível, focando no desenvolvimento de funcionalidades que necessitam de validação para garantir que o projeto esteja no caminho certo e alinhado com as necessidades reais dos usuários e objetivos do negócio.

**Gabarito:** Correto

---

**9. (PROF. DIEGO / INÉDITA – 2024)** No Método MoSCoW, "Won't Have" indica itens que são imperativos para a entrega atual do projeto.

### Comentários:

No Método MoSCoW, "Won't Have" refere-se a itens que não são considerados necessários para a entrega atual do projeto. Esses itens são identificados como de menor prioridade ou como melhorias que podem ser adiadas para versões futuras do projeto. Portanto, ao contrário da afirmação, "Won't Have" indica itens que não são imperativos para a entrega atual.

**Gabarito:** Errado

---

**10. (PROF. DIEGO / INÉDITA – 2024)** A técnica de Valor de Negócio x Risco ajuda a balancear o potencial de retorno com os desafios associados à implementação de itens do backlog.

### Comentários:



A técnica de "Valor de Negócio x Risco" efetivamente ajuda a balancear o potencial de retorno com os desafios associados à implementação de itens do backlog. Ao avaliar itens com base no valor que proporcionam ao negócio contra os riscos de sua implementação, as equipes podem tomar decisões mais informadas sobre quais itens priorizar. Isso assegura que os recursos sejam alocados de maneira eficiente, maximizando o retorno sobre o investimento (ROI) enquanto se gerenciam os riscos de forma proativa.

**Gabarito:** Correto

**11. (PROF. DIEGO / INÉDITA – 2024)** Itens "Could Have" no Método MoSCoW são tratados como prioritários sobre os itens "Should Have".

#### Comentários:

No Método MoSCoW, itens "Could Have" não são tratados como prioritários sobre os itens "Should Have". Na verdade, a prioridade é justamente o oposto: os itens "Should Have" são considerados mais importantes e prioritários do que os itens "Could Have". A hierarquia de prioridades no Método MoSCoW segue a ordem de Must Have (Deve Ter), Should Have (Deveria Ter), Could Have (Poderia Ter), e Won't Have (Não Terá Desta Vez), com os itens "Should Have" tendo prioridade sobre os "Could Have".

**Gabarito:** Errado

**12. (PROF. DIEGO / INÉDITA – 2024)** A avaliação do risco no BUC inclui considerar dependências de outras tarefas ou funcionalidades.

#### Comentários:

Na avaliação do risco no Método BUC (Benefício, Urgência, Custo), de fato inclui-se a consideração de dependências de outras tarefas ou funcionalidades. Essas dependências podem impactar significativamente a execução de um item do backlog, afetando tanto o seu custo quanto a sua viabilidade de implementação dentro do cronograma projetado. Reconhecer e avaliar essas dependências é crucial para uma gestão de risco eficaz e para a priorização informada dos itens do backlog.

**Gabarito:** Correto

**13. (PROF. DIEGO / INÉDITA – 2024)** No Scorecard, o cálculo da pontuação total de cada item é feito sem levar em conta os pesos atribuídos aos critérios.

#### Comentários:





No Scorecard, o cálculo da pontuação total de cada item geralmente leva em conta os pesos atribuídos aos critérios. A ponderação permite que a equipe de projeto reflita a importância relativa de cada critério de acordo com os objetivos estratégicos do projeto ou as prioridades da organização. Ao aplicar pesos diferenciados, a equipe pode garantir que os critérios mais críticos tenham um impacto proporcionalmente maior na pontuação total, facilitando uma priorização mais alinhada com os valores e necessidades do negócio.

**Gabarito:** Errado

**14. (PROF. DIEGO / INÉDITA – 2024)** A técnica de Testes de Suposição ajuda a equipe a aprender e adaptar-se rapidamente, validando suposições importantes por meio de experimentos.

**Comentários:**

A técnica de Testes de Suposição de fato ajuda a equipe a aprender e adaptar-se rapidamente, pois foca na validação de suposições importantes através de experimentos. Este método permite identificar rapidamente quais aspectos do projeto ou produto necessitam de ajustes, reduzindo assim o tempo e os recursos desperdiçados em direções que talvez não sejam viáveis ou valiosas. Ao validar suposições críticas desde cedo, as equipes podem fazer iterações em suas abordagens com base em dados reais e feedback do usuário, promovendo uma cultura de aprendizado contínuo e adaptação ágil.

**Gabarito:** Correto

**15. (PROF. DIEGO / INÉDITA – 2024)** Itens de alto valor e baixo risco são considerados menos prioritários na técnica de Valor de Negócio x Risco.

**Comentários:**

Na técnica de "Valor de Negócio x Risco", itens de alto valor e baixo risco são considerados mais prioritários, não menos. Esta abordagem prioriza itens que oferecem o maior benefício para o negócio com o menor risco associado, maximizando assim o retorno sobre o investimento e minimizando potenciais obstáculos para a implementação. Portanto, itens de alto valor e baixo risco são vistos como oportunidades ideais para alcançar impactos positivos significativos no projeto ou no negócio.

**Gabarito:** Errado

**16. (PROF. DIEGO / INÉDITA – 2024)** Utilizar o Método MoSCoW ajuda na comunicação com as partes interessadas, oferecendo uma visão clara do que esperar na entrega do projeto.

**Comentários:**



Utilizar o Método MoSCoW de fato ajuda na comunicação com as partes interessadas, oferecendo uma visão clara do que esperar na entrega do projeto. Ao categorizar as tarefas em Must have, Should have, Could have e Won't have, a equipe de projeto pode comunicar eficazmente as prioridades e estabelecer expectativas realistas sobre os recursos e funcionalidades que serão entregues. Isso facilita o alinhamento entre a equipe de projeto e as partes interessadas, ajudando a gerenciar expectativas e a focar nos aspectos mais críticos do projeto.

**Gabarito:** Correto

**17. (PROF. DIEGO / INÉDITA – 2024)** Qual das seguintes afirmações melhor descreve a categoria "Must Have" no Método MoSCoW?

- a) Funcionalidades que podem ser adiadas para versões futuras do projeto.
- b) Funcionalidades que agregam valor significativo, mas não são essenciais.
- c) Funcionalidades essenciais sem as quais o projeto não pode ser considerado completo.
- d) Funcionalidades que não serão incluídas na entrega atual do projeto.
- e) Funcionalidades desejáveis que serão incluídas se o tempo e os recursos permitirem.

**Comentários:**

(a) Correto. A categoria "Must Have" no Método MoSCoW refere-se a funcionalidades essenciais sem as quais o projeto não pode ser considerado completo. Estes requisitos são críticos para o lançamento do projeto e não podem ser adiados;

(b) Errado. Funcionalidades que podem ser adiadas para versões futuras do projeto são tipicamente classificadas como "Should Have" ou "Could Have", não "Must Have";

(c) Errado. Funcionalidades que agregam valor significativo, mas não são essenciais, geralmente se enquadram nas categorias "Should Have" ou "Could Have";

(d) Errado. Funcionalidades que não serão incluídas na entrega atual do projeto são classificadas como "Won't Have this time";

(e) Errado. Funcionalidades desejáveis que serão incluídas se o tempo e os recursos permitirem são classificadas como "Could Have".

**Gabarito:** Letra A

**18. (PROF. DIEGO / INÉDITA – 2024)** No contexto do Scorecard para priorização de backlog, como as pontuações são atribuídas a cada item?

- a) Com base em preferências pessoais da equipe de desenvolvimento.
- b) Através da comparação direta com itens semelhantes de outros projetos.



- c) Avaliando cada item contra critérios definidos, com pontuações de acordo com o grau de atendimento.
- d) Utilizando uma abordagem de consenso geral sem critérios específicos.
- e) Atribuindo valores aleatórios e ajustando-os conforme o progresso do projeto.

### Comentários:

(a) Errado. As pontuações no contexto do Scorecard para priorização de backlog não são baseadas em preferências pessoais da equipe de desenvolvimento, mas sim em critérios objetivos pré-definidos;

(b) Errado. Embora a comparação com itens semelhantes de outros projetos possa fornecer insights, as pontuações no Scorecard são atribuídas com base em critérios específicos para o projeto em questão, e não através de comparação direta

(c) Correto. No Scorecard para priorização de backlog, cada item é avaliado contra um conjunto de critérios definidos. Pontuações são atribuídas com base no grau em que cada item atende a esses critérios, possibilitando uma priorização objetiva e justa;

(d) Errado. A abordagem de consenso geral pode ser utilizada em algumas técnicas de priorização, mas no Scorecard, as pontuações são atribuídas com base em critérios específicos, não apenas em consenso geral sem critérios definidos;

(e) Errado. Valores aleatórios não são utilizados na metodologia do Scorecard. As pontuações são cuidadosamente atribuídas com base em como os itens atendem aos critérios estabelecidos, permitindo uma priorização mais objetiva e estratégica.

**Gabarito:** Letra C

**19.(PROF. DIEGO / INÉDITA – 2024)** Qual critério NÃO é comumente utilizado na técnica BUC para avaliação de itens do backlog?

- a) Benefício para o cliente final.
- b) Custo de implementação em termos de recursos.
- c) Urgência baseada em tendências de mercado.
- d) A cor preferida do cliente para a interface do usuário.
- e) Risco associado à implementação do item.

### Comentários:

(a) Errado. O benefício para o cliente final é um critério comumente utilizado na técnica BUC (Benefício, Urgência, Custo) para avaliar itens do backlog, pois foca em maximizar o valor entregue ao cliente;



- (b) Errado. O custo de implementação em termos de recursos é um dos critérios fundamentais na técnica BUC, ajudando a determinar se o retorno justifica o investimento necessário;
- (c) Errado. A urgência baseada em tendências de mercado é outro critério relevante na técnica BUC, uma vez que pode determinar a priorização de itens que aproveitam oportunidades de mercado ou que respondem a necessidades imediatas do negócio;
- (d) Correto. A cor preferida do cliente para a interface do usuário não é um critério utilizado na técnica BUC para avaliação de itens do backlog. A técnica BUC foca em aspectos mais estratégicos e de alto nível, como benefício, urgência e custo, em vez de preferências específicas de design;
- (e) Errado. O risco associado à implementação do item é considerado na técnica BUC, já que pode afetar diretamente o custo e a viabilidade de realizar o item do backlog, bem como seu potencial benefício.

**Gabarito:** Letra D

**20. (PROF. DIEGO / INÉDITA – 2024)** O que a técnica de Testes de Suposição visa validar em um projeto?

- a) A capacidade técnica da equipe de desenvolvimento.
- b) O orçamento total necessário para a conclusão do projeto.
- c) As hipóteses subjacentes aos itens do backlog, especialmente em termos de valor e viabilidade.
- d) O cronograma de entrega para todas as funcionalidades planejadas.
- e) A preferência de cores e design entre os usuários finais.

### Comentários:

- (a) Errado. A técnica de Testes de Suposição não se concentra diretamente na avaliação da capacidade técnica da equipe de desenvolvimento, mas sim na validação das suposições feitas durante o planejamento do projeto;
- (b) Errado. Embora o orçamento seja um componente crucial de qualquer projeto, os Testes de Suposição são mais focados em validar as hipóteses relacionadas aos aspectos do projeto, como valor para o cliente e viabilidade técnica, do que em estimar custos
- (c) Correto. A técnica de Testes de Suposição é utilizada para validar as hipóteses subjacentes aos itens do backlog, especialmente em termos de valor para o cliente e viabilidade técnica ou de negócios. Isso ajuda a garantir que o projeto esteja no caminho certo para entregar valor real e mitigar riscos;



(d) Errado. O foco dos Testes de Suposição não está em validar o cronograma de entrega de todas as funcionalidades planejadas, mas sim em testar a validade das suposições feitas sobre o projeto, o que pode, indiretamente, afetar o cronograma;

(e) Errado. Preferências de cores e design entre os usuários finais podem ser importantes, mas os Testes de Suposição visam validar hipóteses mais críticas para o sucesso do projeto, como a necessidade do produto e sua viabilidade, não aspectos específicos de design.

**Gabarito:** Letra C

**21. (PROF. DIEGO / INÉDITA – 2024)** Na técnica de priorização de Valor de Negócio x Risco, como os itens são classificados?

- a) Itens de alto risco são sempre priorizados, independentemente de seu valor de negócio.
- b) Itens de baixo valor de negócio e alto risco são priorizados para maximizar o desafio.
- c) Itens de baixo valor de negócio e baixo risco são os primeiros a serem desenvolvidos.
- d) Itens são classificados aleatoriamente para incentivar a inovação.
- e) Itens de alto valor de negócio e baixo risco são priorizados para entrega.

#### Comentários:

(a) Errado. A técnica de Valor de Negócio x Risco não prioriza itens de alto risco automaticamente; a priorização depende de uma avaliação conjunta do valor de negócio e do risco associado;

(b) Errado. Itens de baixo valor de negócio e alto risco geralmente não são priorizados, pois essa combinação não é ideal para maximizar o retorno sobre o investimento ou mitigar riscos significativos

(c) Errado. Itens de baixo valor de negócio e baixo risco geralmente não são os primeiros a serem desenvolvidos, pois essa abordagem não maximiza o valor para o negócio nem efetivamente utiliza recursos para mitigar riscos importantes;

(d) Errado. A classificação não é feita de maneira aleatória; ela é cuidadosamente determinada com base no valor de negócio e no risco de cada item para garantir uma tomada de decisão estratégica;

(e) Correto. Itens de alto valor de negócio e baixo risco são priorizados para entrega, pois essa combinação maximiza o retorno sobre o investimento e minimiza os riscos, alinhando-se efetivamente com os objetivos estratégicos do negócio.

**Gabarito:** Letra E

**22. (PROF. DIEGO / INÉDITA – 2024)** Qual é o objetivo principal do Método MoSCoW na priorização de requisitos de projeto?



- a) Identificar quais requisitos são essenciais para o sucesso do projeto e quais podem ser adiados.
- b) Eliminar completamente o risco associado a cada requisito do projeto.
- c) Assegurar que todos os requisitos sejam tratados como igualmente importantes.
- d) Determinar o orçamento necessário para a implementação de cada requisito.
- e) Estabelecer uma sequência cronológica exata para o desenvolvimento de cada requisito.

### Comentários:

(a) Correto. O Método MoSCoW é uma técnica de priorização usada para determinar a importância de cada requisito de projeto, identificando quais são essenciais (Must have) para o sucesso do projeto e quais podem ser adiados (Should have, Could have, Won't have this time). Este método ajuda a focar nos requisitos mais críticos;

(b) Errado. O Método MoSCoW não tem como objetivo eliminar completamente o risco, mas priorizar requisitos com base em sua importância e urgência, não sua associação ao risco;

(c) Errado. Este método especificamente não trata todos os requisitos como igualmente importantes; pelo contrário, classifica os requisitos em categorias de prioridade;

(d) Errado. Determinar o orçamento necessário para a implementação de cada requisito não é o objetivo principal do Método MoSCoW. Este método foca na priorização de requisitos, não no custo diretamente;

(e) Errado. Estabelecer uma sequência cronológica exata para o desenvolvimento de cada requisito não é o foco do Método MoSCoW. O método ajuda a definir a prioridade dos requisitos, mas não especifica a ordem exata de desenvolvimento.

**Gabarito:** Letra A



## LISTA DE QUESTÕES

1. **(PROF. DIEGO / INÉDITA – 2024)** O Método MoSCoW pode ser usado para identificar funcionalidades que são essenciais para o lançamento inicial de um produto.
2. **(PROF. DIEGO / INÉDITA – 2024)** Itens classificados como "Should Have" no Método MoSCoW são essenciais para o sucesso do projeto e não podem ser omitidos.
3. **(PROF. DIEGO / INÉDITA – 2024)** No Scorecard para priorização de backlog, a atribuição de pontuações é feita sem considerar critérios pré-definidos.
4. **(PROF. DIEGO / INÉDITA – 2024)** A técnica de Scorecard permite comparações objetivas entre diferentes tarefas ao atribuir valores numéricos a cada item do backlog.
5. **(PROF. DIEGO / INÉDITA – 2024)** No BUC, itens com maior urgência sempre recebem a prioridade mais alta, independentemente de seu benefício ou custo.
6. **(PROF. DIEGO / INÉDITA – 2024)** O critério de Benefício no BUC avalia o impacto positivo que a implementação de um item do backlog trará para o projeto.
7. **(PROF. DIEGO / INÉDITA – 2024)** A técnica de Testes de Suposição foca exclusivamente na viabilidade técnica das funcionalidades, ignorando seu valor para os usuários.
8. **(PROF. DIEGO / INÉDITA – 2024)** Testes de Suposição priorizam funcionalidades para desenvolvimento com base na necessidade de validar hipóteses críticas para o sucesso do projeto.
9. **(PROF. DIEGO / INÉDITA – 2024)** No Método MoSCoW, "Won't Have" indica itens que são imperativos para a entrega atual do projeto.
10. **(PROF. DIEGO / INÉDITA – 2024)** A técnica de Valor de Negócio x Risco ajuda a balancear o potencial de retorno com os desafios associados à implementação de itens do backlog.
11. **(PROF. DIEGO / INÉDITA – 2024)** Itens "Could Have" no Método MoSCoW são tratados como prioritários sobre os itens "Should Have".
12. **(PROF. DIEGO / INÉDITA – 2024)** A avaliação do risco no BUC inclui considerar dependências de outras tarefas ou funcionalidades.
13. **(PROF. DIEGO / INÉDITA – 2024)** No Scorecard, o cálculo da pontuação total de cada item é feito sem levar em conta os pesos atribuídos aos critérios.





- 14. (PROF. DIEGO / INÉDITA – 2024)** A técnica de Testes de Suposição ajuda a equipe a aprender e adaptar-se rapidamente, validando suposições importantes por meio de experimentos.
- 15. (PROF. DIEGO / INÉDITA – 2024)** Itens de alto valor e baixo risco são considerados menos prioritários na técnica de Valor de Negócio x Risco.
- 16. (PROF. DIEGO / INÉDITA – 2024)** Utilizar o Método MoSCoW ajuda na comunicação com as partes interessadas, oferecendo uma visão clara do que esperar na entrega do projeto.
- 17. (PROF. DIEGO / INÉDITA – 2024)** Qual das seguintes afirmações melhor descreve a categoria "Must Have" no Método MoSCoW?
- a) Funcionalidades que podem ser adiadas para versões futuras do projeto.
  - b) Funcionalidades que agregam valor significativo, mas não são essenciais.
  - c) Funcionalidades essenciais sem as quais o projeto não pode ser considerado completo.
  - d) Funcionalidades que não serão incluídas na entrega atual do projeto.
  - e) Funcionalidades desejáveis que serão incluídas se o tempo e os recursos permitirem.
- 18. (PROF. DIEGO / INÉDITA – 2024)** No contexto do Scorecard para priorização de backlog, como as pontuações são atribuídas a cada item?
- a) Com base em preferências pessoais da equipe de desenvolvimento.
  - b) Através da comparação direta com itens semelhantes de outros projetos.
  - c) Avaliando cada item contra critérios definidos, com pontuações de acordo com o grau de atendimento.
  - d) Utilizando uma abordagem de consenso geral sem critérios específicos.
  - e) Atribuindo valores aleatórios e ajustando-os conforme o progresso do projeto.
- 19. (PROF. DIEGO / INÉDITA – 2024)** Qual critério NÃO é comumente utilizado na técnica BUC para avaliação de itens do backlog?
- a) Benefício para o cliente final.
  - b) Custo de implementação em termos de recursos.
  - c) Urgência baseada em tendências de mercado.
  - d) A cor preferida do cliente para a interface do usuário.
  - e) Risco associado à implementação do item.
- 20. (PROF. DIEGO / INÉDITA – 2024)** O que a técnica de Testes de Suposição visa validar em um projeto?
- a) A capacidade técnica da equipe de desenvolvimento.
  - b) O orçamento total necessário para a conclusão do projeto.
  - c) As hipóteses subjacentes aos itens do backlog, especialmente em termos de valor e viabilidade.



- d) O cronograma de entrega para todas as funcionalidades planejadas.
- e) A preferência de cores e design entre os usuários finais.

**21. (PROF. DIEGO / INÉDITA – 2024)** Na técnica de priorização de Valor de Negócio x Risco, como os itens são classificados?

- a) Itens de alto risco são sempre priorizados, independentemente de seu valor de negócio.
- b) Itens de baixo valor de negócio e alto risco são priorizados para maximizar o desafio.
- c) Itens de baixo valor de negócio e baixo risco são os primeiros a serem desenvolvidos.
- d) Itens são classificados aleatoriamente para incentivar a inovação.
- e) Itens de alto valor de negócio e baixo risco são priorizados para entrega.

**22. (PROF. DIEGO / INÉDITA – 2024)** Qual é o objetivo principal do Método MoSCoW na priorização de requisitos de projeto?

- a) Identificar quais requisitos são essenciais para o sucesso do projeto e quais podem ser adiados.
- b) Eliminar completamente o risco associado a cada requisito do projeto.
- c) Assegurar que todos os requisitos sejam tratados como igualmente importantes.
- d) Determinar o orçamento necessário para a implementação de cada requisito.
- e) Estabelecer uma sequência cronológica exata para o desenvolvimento de cada requisito.



## GABARITO

1. CORRETO
2. ERRADO
3. ERRADO
4. CORRETO
5. ERRADO
6. CORRETO
7. ERRADO
8. CORRETO
9. ERRADO
10. CORRETO
11. ERRADO
12. CORRETO
13. ERRADO
14. CORRETO
15. ERRADO
16. CORRETO
17. LETRA A
18. LETRA C
19. LETRA D
20. LETRA C
21. LETRA E
22. LETRA A



# ESSA LEI TODO MUNDO CONHECE: PIRATARIA É CRIME.

Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.



**1** Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.



**2** Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).



**3** Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).



**4** Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).



**5** Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).



**6** Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).



**7** Concurseiro(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.



**8** O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.