

Eletrônico



Estratégia
CONCURSOS

Aula

Engenharia de Software II (ISS-Manaus (Técnico de Tecnologia da Informação) – Pós-Edital)

Professor: Diego Carvalho, Equipe Informática e TI

SUMÁRIO	PÁGINA
Apresentação	01
- Engenharia de Requisitos	02
Lista de Exercícios Comentados	65
Gabarito	82





Cara, esse é um conceito bem tranquilo de entender! *Vamos passar para a vida real?* É época de matrícula na faculdade e você cursa o 2º Semestre de Ciência da Computação, mas você quer pegar uma disciplina lá do 8º Semestre! No entanto, há um pequeno detalhe! **Essa disciplina provavelmente contém diversos pré-requisitos que você tem que cumprir para fazê-la.**

Querem outro exemplo? **Vamos para a área jurídica! Algum de vocês aí quer mudar o Brasil e decide se candidatar à presidência da república!** Espera um pouco... para tal, devem ser cumpridos alguns requisitos: ser filiado a um partido político; ter domicílio eleitoral na circunscrição; ter realizado o alistamento eleitoral (caso seja homem); estar em pleno exercício dos direitos políticos; e ser brasileiro nato.

Pois é... mesmo cumprindo todos esses requisitos, ainda falta mais um: ter idade mínima de 35 anos! Satisfeitos esses requisitos, você – revolucionário que irá mudar nossa nação – poderá ser candidato à presidência. *Bacana, pessoal?* Na vida real fica bem tranquilo de compreender o que é um requisito! É natural, pois faz parte do nosso cotidiano.

O verbo *requisitar* é o mesmo que pedir, solicitar, exigir algo que seja condição necessária para alguma finalidade específica. Grosso modo, o requisito é uma condição ou exigência indispensável para um fim determinado. **Vocês conhecem o RUP? Ele conceitua requisito como uma condição ou capacidade com a qual o sistema deve estar de acordo.**

Na Engenharia, costuma-se definir um requisito como uma propriedade ou um comportamento que um produto ou serviço deve atender. Na Engenharia de Software, dizemos que um requisito é a 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.

Ou, também, uma condição, característica ou capacidade, determinada no universo das necessidades do negócio do usuário. Bem, esses conceitos são todos muito parecidos e é fácil de associar a situações comuns em nossas vidas. **Atenção, porque esse é um tema clássico de provas discursivas!** Portanto, tenham esses conceitos perfeitamente guardados em suas memórias. *Ok? ;)*



CLASSIFICAÇÃO DE REQUISITOS

Vamos falar um pouco agora sobre as classificações de requisitos – existem várias! Alguns dos problemas que ocorrem durante o processo de engenharia de requisitos são resultado da **falha em se fazer uma clara separação entre os níveis de abstração da descrição dos requisitos**. Galera, usuários são leigos e eles não têm a menor obrigação de entender detalhes técnicos. Dito isso, vamos às classificações!

CLASSIFICAÇÃO QUANTO AO NÍVEL DE ABSTRAÇÃO

- **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.
 - **Exemplo:** O software deve fornecer um meio de representar e acessar arquivos externos criados por outras ferramentas.
- **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.
 - **Exemplo:** O usuário deve dispor de recursos de tecnologia assistiva para acessar dados de bancos externos.

Professor, isso é só para complicar a minha vida? Não, pessoal! Isso tem a sua utilidade! Deve-se escrever os requisitos em diferentes níveis de detalhamento, tendo em vista que os leitores os utilizam de maneiras diferentes. Usuários geralmente não dão a mínima para como o sistema será implementado. Já os leitores de requisitos de sistema necessitam de detalhes mais exatos e precisos.

CLASSIFICAÇÃO QUANTO À QUALIDADE

O **Quality Function Deployment (QFD)** é uma técnica de gestão da qualidade que traduz as necessidades do cliente para requisitos técnicos. O QFD concentra-se em maximizar a satisfação do cliente por meio do processo de engenharia de software. Para tal, enfatiza o entendimento do que é valioso para o cliente e emprega esses valores ao longo do processo de engenharia.



- **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 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.
- **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.

CLASSIFICAÇÃO QUANTO À EVOLUÇÃO

- **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.
- **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:
 - **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 o sistema e esses requisitos são uma consequência (olha a dica!) 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.

CLASSIFICAÇÃO QUANTO À FUNCIONALIDADE

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, isso vai cair na sua prova objetiva e talvez até na sua prova discursiva, mas fiquem tranquilos porque é fácil ;)

Requisitos Funcionais: **são 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.** Depende do tipo de software, dos usuários esperados e do tipo de sistema onde o software será implantado. Eles fazem parte da arquitetura do aplicativo de um sistema.

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 nos serviços ou funções oferecidas pelo sistema*. Não estão diretamente relacionados às funções específicas do sistema, mas às gerais e podem incluir restrições de tempo, processo de desenvolvimento e restrições impostas por padrões. Podem ser mais críticos que os funcionais e sempre devem ser verificáveis. Eles fazem parte da arquitetura técnica de um sistema.

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. Por 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* na 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.

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.

IMPORTANTE

Os Requisitos de Domínio são um pouco problemáticos e, muitas vezes, inúteis. Curiosamente, na última edição do livro do Sommerville, deu-se bem menos destaque. Para quem ainda não os entendeu, vejamos: no sistema de uma Biblioteca, pode existir o requisito "Permitir adicionar livro"; no sistema de uma Locadora, pode existir o requisito "Permitir adicionar filme"; no sistema de uma loja de Instrumentos Musicais, pode existir o requisito "Permitir adicionar guitarra".

Vocês percebem que, de modo geral, é tudo a mesma coisa? Pois é! Agora imaginem o requisito "Permitir que o valor do Título Público NTN-B possa ser multiplicado pelo valor da Taxa Selic corrente quando essa for menor que 0,5% ao mês". Ora, esse requisito dificilmente poderá ser aplicado em algum outro contexto que não seja aquele específico de um sistema/domínio de venda de títulos da dívida pública, portanto, trata-se de um Requisito de Domínio.

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. Tratam-se de qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras. Normalmente estes requisitos são descritos de maneira informal. **Galera, em geral, afetam a arquitetura do sistema.**

CLASSIFICAÇÃO QUANTO À ORIGEM

Agora vamos partir para outra classificação um pouco menos importante! Pesquisadores observaram que **os requisitos não-funcionais também podem 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 dividem também em várias outras, como mostra a imagem abaixo. Cabe salientar que essa classificação está em conformidade com a 9ª Edição do 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.
REQUISITOS DE DESEMPENHO	O sistema deverá ser capaz de processar oitocentas requisições por segundo.
REQUISITOS DE ESPAÇO (OU ARMAZENAMENTO)	O sistema deverá ocupar, no máximo, 80Mb da memória interna do dispositivo.
REQUISITOS DE USABILIDADE	Os usuários deverão operar todas as funcionalidades do sistema após 2 horas de treino.
REQUISITOS DE SEGURANÇA	O sistema não deve permitir a ativação simultânea de mais de três sinais de alarme.
REQUISITOS ÉTICOS	O sistema não apresentará aos usuários quaisquer dados de natureza confidencial de outrem.
REQUISITOS DE IMPLEMENTAÇÃO	A interface de usuário deve ser implementada em HTML e não se deve utilizar Applets de Java.

REQUISITOS X REGRAS DE NEGÓCIO

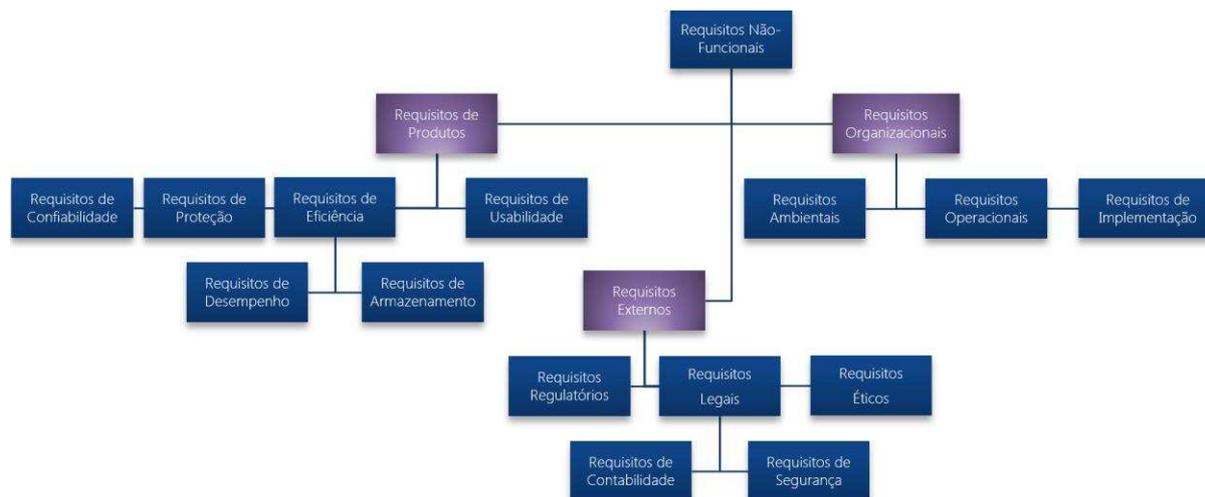
Requisitos tratam do funcionamento de um sistema e Regras de Negócio tratam do funcionamento de um negócio, sendo independentes de um sistema específico. *Vamos ver um exemplo no contexto de um caixa eletrônico?*

1. O Sistema deve sacar da conta a quantia determinada pelo usuário – trata-se de um Requisito Funcional, visto que é uma funcionalidade oferecida pelo sistema.
2. O Limite de Saque é de R\$1000 por dia – trata-se de uma Regra de Negócio, visto que é uma regra do banco e não só do sistema.

Eventualmente, há uma confusão entre Requisitos Funcionais e Requisitos Não-Funcionais. Por exemplo:

1. O Sistema deve conter uma política de controle de acesso – trata-se de um Requisito Não-Funcional, visto que isso não é uma funcionalidade em si.
2. O Sistema não deve permitir que usuários com Perfil X alterem dados – trata-se de um Requisito Funcional, visto que é uma restrição à funcionalidade e, não, ao sistema.





ENGENHARIA DE REQUISITOS

Pessoal, considerem a seguinte hipótese: vocês fazem uma excelente prova objetiva e partem para a prova discursiva. O primeiro item pede a definição de Requisito! Bem, vocês abrem um sorriso e respondem! O segundo item pede a definição de Engenharia de Requisitos. **Bem, agora não tem mais sorriso! Esse é um conceito que todo mundo sabe, mas que é difícil de colocar no papel.**

Porém, vocês enrolam e conseguem responder. Aí vem o terceiro item, que pede as fases do processo de engenharia de requisitos! *E aí, quantos de vocês conseguiriam responder? Estão vendo como é complicado?* **Então vamos entender tudo isso... esse termo 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 90 com a publicação de um tutorial da IEEE e a criação de uma série de conferências sobre esse assunto. *Ok, mas qual é a definição de Engenharia de Requisitos? Ora, ela pode ser definida como* **uma abordagem sistemática para a formulação, análise, documentação e manutenção de requisitos de um sistema.** Fácil, né?! ;)

Pessoal, 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. Sabe-se que muitos engenheiros se perguntam como assegurar que eles especificaram um sistema exatamente de acordo com as necessidades dos clientes.

Bem, não há uma resposta perfeita para esse questionamento, contudo **um sólido processo de engenharia de requisitos é capaz de encontrar a melhor solução viável no momento.** Percebam que é praticamente impossível satisfazer absolutamente todas as expectativas dos clientes, na medida em que requisitos podem ser bastante complexos e voláteis.

Na prática, todos 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.





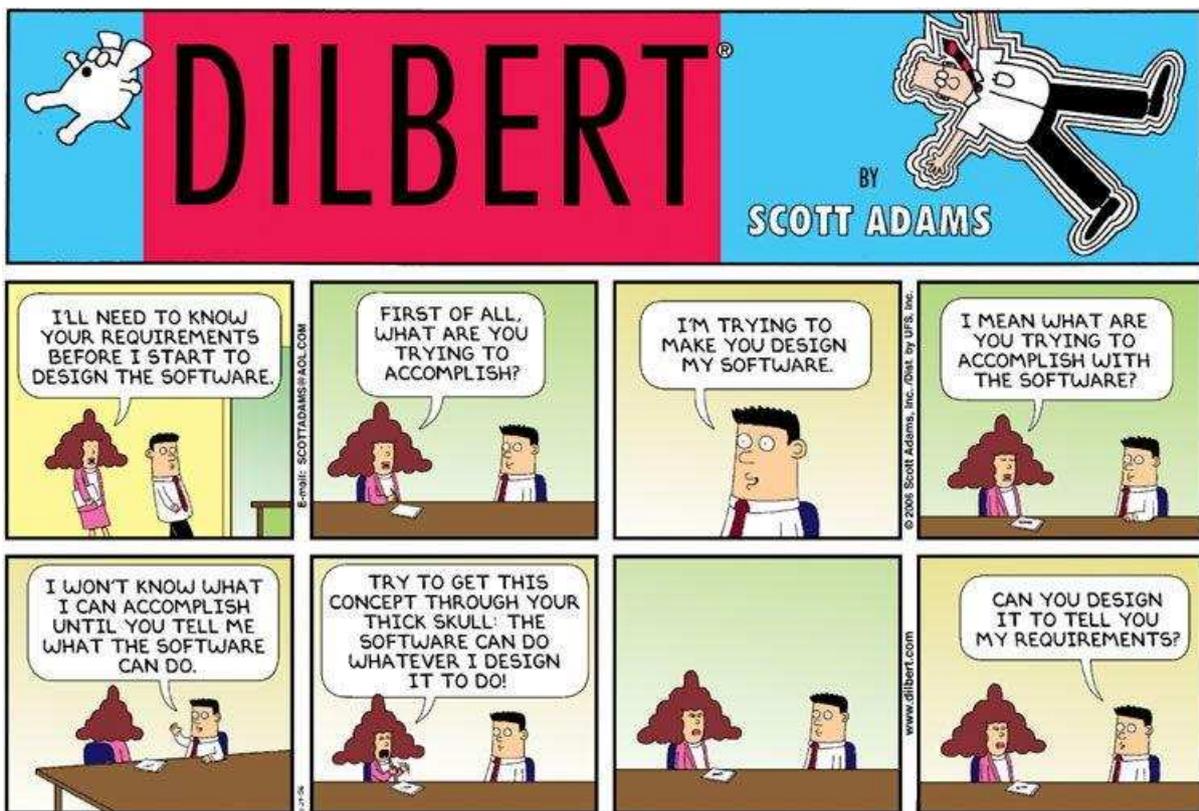
O Engenheiro de Software da IBM, Fred Brooks diz: "A parte mais árdua na construção de um sistema de software consiste em decidir o que construir! Nenhuma outra fase do processo de desenvolvimento compromete tanto o resultado final do sistema se feita errada e nenhuma outra dificulta tanto as correções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



A Engenharia de Requisitos fornece um mecanismo adequado para compreender o que o cliente deseja, para analisar as necessidades, para avaliar a viabilidade, para negociar uma solução razoável, para especificar um 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 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 mais cobrado em provas nesse quesito**, porém não custa nada ver também as fases do Pressman abaixo:

1. **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.
2. **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?
3. **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.
4. **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.
5. **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.
6. **Validação:** 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.
7. **Gestão:** conjunto de atividades que auxiliam a equipe de projeto a identificar, controlar e rastrear requisitos e mudanças nos requisitos a qualquer momento¹. Para projetos de grande porte, é uma fase essencial na medida em que mudanças em um requisito podem afetar diversos outros requisitos.

¹ Para tal, recomenda-se utilizar tabelas de rastreamento de características, fontes, dependências, subsistemas e interfaces.



POR SOMMERVILLE	POR PRESSMAN
Estudo de Viabilidade	Concepção
Elicitação e Análise de Requisitos - Obtenção de Requisitos - Classificação e Organização - Priorização e Negociação - Documentação de Requisitos	Levantamento
	Elaboração
	Negociação
	Especificação
Validação	Validação
Gestão	Gestão

Muito bem! Se vocês respondessem a prova discursiva com essas fases, estaria **perfeitamente correto**. Acima podemos ver uma comparação entre as fases de acordo com ambos os autores. Lembrando que essas fases buscam produzir um documento de requisitos que especifique um sistema e satisfaça os requisitos dos stakeholders. Entendido? ;)

Bem, chegou a hora de eu ganhar alguns inimigos! **Por muito tempo, eu deixei isso fora da aula para evitar que isso acontecesse, mas eu cheguei à conclusão de que pode ajudar muita gente – além de ser apenas uma bobagem!** Bem, quando eu estudava esse assunto, eu criei mnemônicos para decorar as fases de cada autor. Espero que ajudem vocês a decorarem.

FASES PRESSMAN:
1 - CENAS LAMENTÁVEIS EM NOVO EMPATE DO VASCO DA GAMA
2 - CONSEGUI LEVANTAR ELAINE, NEM ESPEREI VALDIR GESTICULAR

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





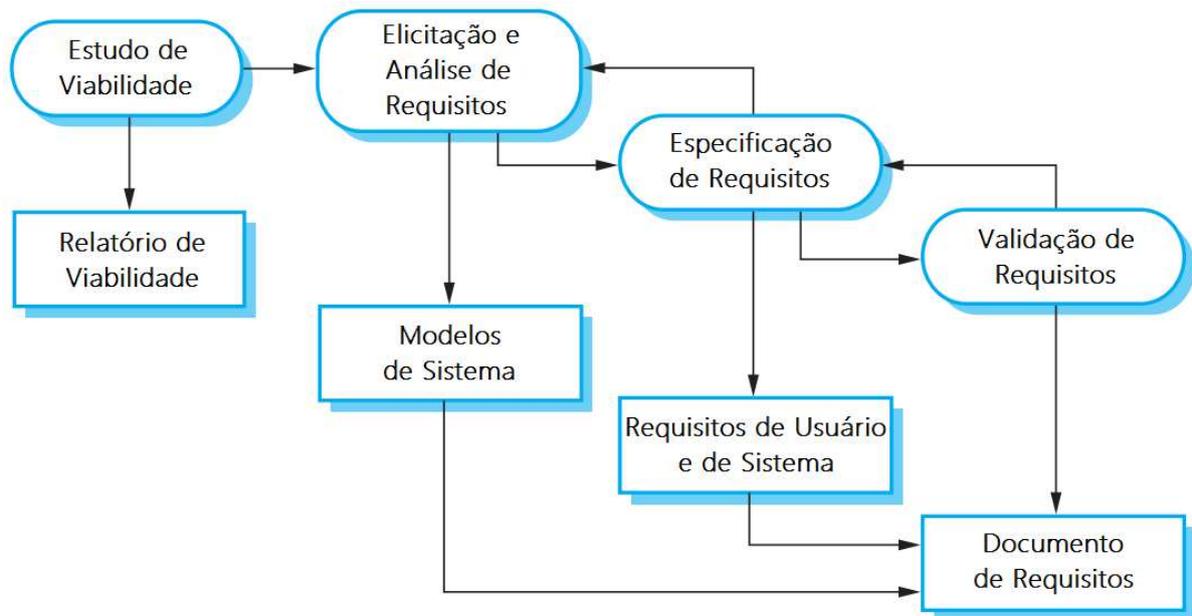
FASES SOMMERVILLE: ESTÃO VIABILIZANDO ELIANA ESPECIALMENTE NO VASCO DA GAMA

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

Vascaínos, peço que não se sintam ofendidos. Isso é apenas uma maneira que eu encontrei para decorar tempos atrás e que efetivamente me ajudou muito. Caso vocês conheçam outros mnemônicos para decorar essas fases, eu posso trocá-los. Estou colocando-os aqui apenas para ajudá-los, caso haja muitas reclamações, eu posso retirá-los. Prosseguindo...

A imagem abaixo 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 envolve todas as anteriores e também é de suma importância: Gerenciamento de Requisitos.





Sommerville afirma que o objetivo da engenharia de requisitos é criar e manter um documento de requisitos de sistema. As fases são: avaliação de se o sistema é útil para a empresa (Estudo de Viabilidade); obtenção de requisitos (Elicitação e Análise); conversão desses requisitos em alguma forma-padrão (Especificação); e verificação de se os requisitos realmente definem o sistema que o cliente deseja (Validação).

ESTUDO DE VIABILIDADE

É a 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 a tomada de decisão** quanto a prosseguir para uma análise mais detalhada ou não.

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. **A fase seguinte utiliza as informações do estudo de viabilidade no levantamento de requisitos.**

Ele deve ser um estudo curto e bem focado, devendo ser realizado no início do processo de engenharia de requisitos e, ao final, deverá entregar um relatório de viabilidade. Além disso, o Estudo de Viabilidade deve responder às três questões principais 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?**

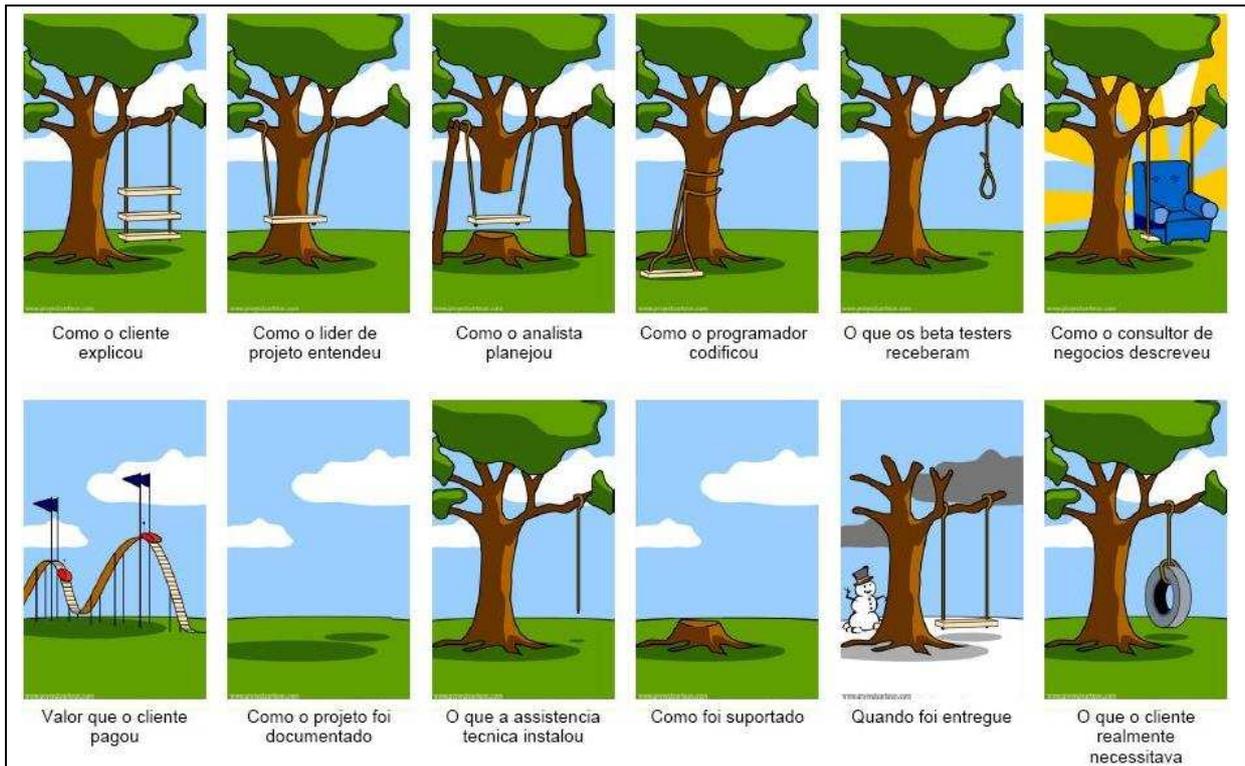
As respostas para essas três questões não é algo simples de ser obtido, visto que geralmente a empresa não tem a exata definição de seus objetivos. Para ajudar nesta definição, questões podem ser levantadas e, com as respostas em mãos, pode-se concluir se o desenvolvimento do sistema deve prosseguir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

- 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?**



ELICITAÇÃO E ANÁLISE DE REQUISITOS

É o **processo de levantamento e derivação de requisitos de sistema** através da observação de sistemas existentes, discussões com usuários potenciais e compradores, análise de tarefas, etc. 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.



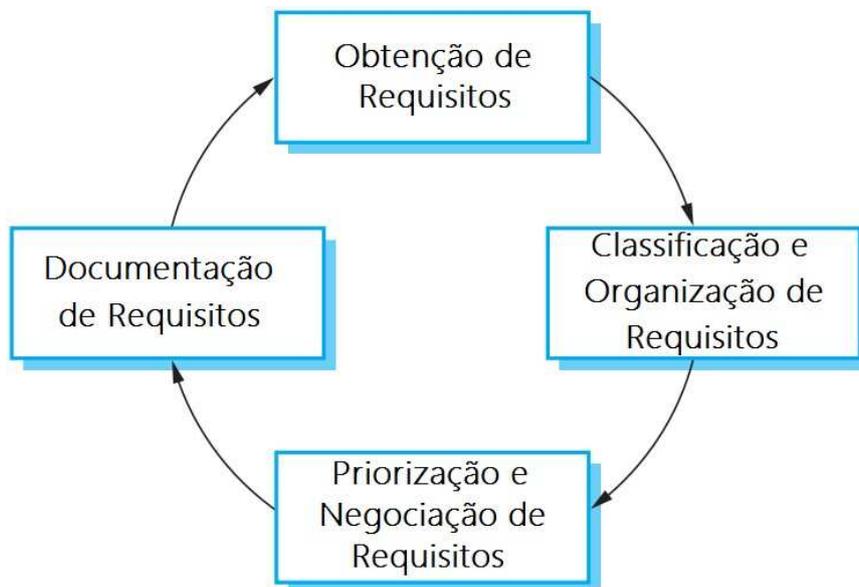
Elicitar é o mesmo que descobrir, identificar, deduzir, extrair, evocar, obter informações sobre uma questão específica. Para tal, os engenheiros de software trabalham com os clientes e usuários finais para aprender sobre o domínio da aplicação, quais serviços o sistema deve fornecer, o desempenho esperado, restrições de hardware, entre outros.



Pessoal, por que dizem por aí que a elicitação e compreensão de requisitos é tão difícil? Ora, muitas vezes, **os stakeholders não sabem o que querem**; outras vezes, deixam dados de seu domínio implícitos por pensarem que são óbvios; há, também, casos em que os requisitos são conflitantes; além de fatores políticos, econômicos e de negócio que

podem influenciar o levantamento de requisitos do sistema. As principais atividades do processo de eliciação 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.



Pessoal, um dos temas campeões em prova se refere às técnicas de eliciação. *Professor, são realmente necessárias técnicas para levantar requisitos? Sim, pessoal!* Como eu já repeti diversas vezes, levantar requisitos é uma tarefa árdua. Portanto, para ajudar a **assegurar uma cobertura ampla dos requisitos**, utilizam-se as seguintes técnicas (sublinhei as que mais caem 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 das respostas a essas questões.**

As entrevistas podem ser abertas, quando não há um roteiro predefinido ou podem ser fechadas, quando há um roteiro pronto. **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; 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 elicitação de conhecimentos sobre os requisitos e as restrições organizacionais, pois existem relacionamentos sutis de poder e influência entre os stakeholders.

As estruturas organizacionais públicas raramente coincidem com a realidade da tomada de decisões na organização, mas os entrevistados podem não querer revelar a estrutura real, em lugar da teórica, a um estranho. **Em geral, a maioria das pessoas é relutante em discutir questões políticas e organizacionais que podem afetar os requisitos.**

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 elicitação.



O valor da etnografia está na ajuda que presta aos analistas para descobrir os requisitos implícitos de sistema que refletem os processos reais, e não os formais, com os quais as pessoas estão envolvidas. **As pessoas frequentemente consideram muito difícil articular detalhes de seu trabalho rotineiro, na medida em que isso é secundário para elas.**

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 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 os detalhes desses cenários. **Os cenários podem ser escritos na forma de textos, complementados por diagramas, imagens de computador etc.** Como alternativa, pode ser adotada uma abordagem mais estruturada, como cenários de eventos ou casos de uso.

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 grande de pessoas para extrair as mesmas informações. 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 dos *stakeholders* que melhor representam a organização e o contexto em que o sistema será usado, obtendo assim um conjunto de requisitos bem definidos. **Por ser realizado por convocação por dia e horário, pode ocasionar problemas no presenciais dos *stakeholders*.**

6. Brainstorming (Tempestade de Ideias)

É uma abordagem de elicitación 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, essa técnica democratiza a participação de membros do grupo. Uma desvantagem é que 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:

Coleta 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 dinâmicas de grupo, técnicas visuais, processos racionais e documentação. **É bastante interativa e promove a participação ativa dos envolvidos.**

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, etc. **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 outros métodos.

10. Reúso de Requisitos:

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.** Assim, eles têm chances maiores de serem compreendidos pelos *stakeholders*.

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 pela Metodologia XP, nada mais é do que uma história contada na linguagem do usuário final 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>".*

12. Participação Ativa de Usuários:

Técnica que permite a incorporação dos usuários 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, etc.

14. Interpretação de Papeis

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

Trata-se de um grupo de discussão informal e de tamanho reduzido (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. No entanto, exige facilitador/moderador com experiência para conduzir o grupo. Além disso, ele 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 elicitare 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):



Essa técnica 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.

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; etc).

Bem, falamos um bocado sobre a Obtenção de Requisitos. Agora vamos falar brevemente do restante das atividades de elicitação e análise de requisitos. **Começemos pela Classificação e Organização.** Pessoal, levantaram-se vários requisitos para o sistema. *Vocês se lembram da quantidade de classificações que nós estudamos, certo?* Ela será útil neste momento!

Agora os diversos requisitos levantados serão agrupados de acordo com suas características comuns e serão organizados em conjuntos coerentes. Além disso, **o analista de requisitos se preocupará em fazer várias verificações de conflito, consistência, omissão e ambiguidade,** de modo que haja um mínimo de erros para a fase seguinte.

Então, chegamos à atividade de priorização e negociação. Nessa fase, **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.

Enfim, chegamos à documentação de requisitos. Esse estágio tem simplesmente a função de **documentar formalmente ou informalmente os requisitos levantados, classificados, organizados, priorizados e negociados,** de modo que possam ser utilizados para auxiliar as próximas obtenções de requisitos. Já se obtém um esboço do documento de requisitos.



ESPECIFICAÇÃO DE REQUISITOS

Atividade de traduzir as informações coletadas durante a atividade de elicitação e análise em um documento que define um conjunto de requisitos. De fato, é o processo de escrever os requisitos de usuário e requisitos de sistema em um artefato chamado Documento de Requisitos². 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. **Importante: requisitos do usuário devem descrever os requisitos funcionais e/ou não-funcionais, de modo que eles sejam compreensíveis pelos usuários do sistema que não possuem conhecimentos técnico detalhado.**

Deve-se descrever os requisitos de usuário em linguagem simples (tabelas, formulários e diagramas intuitivos), especificando somente características externas, evitando características do projeto de sistema, i.e., não devemos usar jargões de software. **Já os requisitos de sistemas 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, **inicialmente 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.

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.

² 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).



VALIDAÇÃO DE REQUISITOS

Essa atividade verifica 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 de requisitos, fazendo uma mudança de sistema, é muito maior do que a correção de erros de projeto e de 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 requisito. 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 genéricos 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.**

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 o objetivo 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.

De acordo com Stephen R. Schach, superficialmente, 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. **Mas a diferença vai muito além. 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, elicitare 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:

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, i.e.,** visa assegurar que as necessidades do cliente estão sendo atendidas por tais requisitos. *Entenderam 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:



1. **Verificações 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.
2. **Verificações 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.
3. **Verificações 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.
4. **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.
5. **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.***



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 preciso 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 de sistema. *Entenderam isso?*

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.

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, etc.

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, 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, tomam-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.

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?*





1. (FCC - 2012 - TST - Analista Judiciário - Análise de Sistemas) 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 as 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: C

2. (FCC - 2012 - TCE-AP - Analista de Controle Externo - Tecnologia da Informação)
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:

Requisitos Funcionais: *são 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.* Depende do tipo de software, dos usuários esperados e do tipo de sistema onde o software será implantado. Eles fazem parte da arquitetura do aplicativo de um sistema.

Requisitos Não-Funcionais: *são restrições nos serviços ou funções oferecidas pelo sistema.* Não estão diretamente relacionados às funções específicas do sistema, mas às gerais e podem incluir restrições de tempo, processo de desenvolvimento e restrições impostas por padrões. Podem ser mais críticos que os funcionais e sempre devem ser verificáveis. Eles fazem parte da arquitetura técnica de um sistema.

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.



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

Gabarito: A

3. (FCC - 2011 - INFRAERO - Analista de Sistemas - Gestão de TI – 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: C

4. (FCC - 2010 - BAHIA GÁS - Analista de Processos Organizacionais - Análise de Sistemas) 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:

Requisitos Não-Funcionais: são restrições nos serviços ou funções oferecidas pelo sistema. Não estão diretamente relacionados às funções específicas do sistema, mas às gerais e podem incluir restrições de tempo, processo de desenvolvimento e restrições impostas por padrões. Podem ser mais críticos que os funcionais e sempre devem ser verificáveis. Eles fazem parte da arquitetura técnica de um sistema.

Conforme vimos em aula, é um Requisito Não-Funcional!

Gabarito: A



5. (FCC - 2009 – SEFAZ/SP - Analista de Sistemas) *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: A

6. (FCC - 2010 – DPE/SP - Analista de Sistemas) 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, i.e., um documento de requisitos de sistema com apenas requisitos funcionais. (c) Correto! Ele se divide nessas três categorias.

Gabarito: C

7. (FCC - 2009 - TRT - 3ª Região (MG) - Analista Judiciário - Tecnologia da Informação) 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: E

8. (FCC - 2010 – DPE/SP - Analista de Sistemas - III) 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, i.e., 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, i.e., 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: A

9. (FCC - 2013 – SEFAZ/SP – Analista de Sistemas) 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, i.e., 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.

Gabarito: C

10. (FCC - 2013 – MPE/MA – Analista de Sistemas) 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 podem ser divididos em Requisitos Ambientais, Operacionais e de Desenvolvimento (ou Implementação).

Gabarito: B

11. (FCC - 2012 - MPE-PE - Analista Ministerial - Informática) 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: A

12. (FCC - 2012 - TRE-CE - Analista Judiciário - Análise de Sistemas) 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!



13. (FCC - 2012 - TST - Analista Judiciário - Análise de Sistemas) 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) Não é durante a Verificação, mas Validação de Software; (b) Não são as necessidades da Equipe de Desenvolvimento, mas dos usuários; (c) Perfeito, é exatamente isso; (d) Estão invertidos, custo e confiabilidade são RNF e Casos de Uso são RF; (e) Isso é responsabilidade do Gerente de Projetos e, não, do Gerente de Requisitos.

14. (FCC - 2012 - TJ-PE - Analista Judiciário - Análise de Sistemas) Na engenharia de requisitos trata-se de uma técnica de eliciação 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:

Brainstorming (Tempestade de Ideias)

É uma abordagem de elicitación 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.**

Conforme vimos em aula, trata-se da Técnica de Brainstorming.

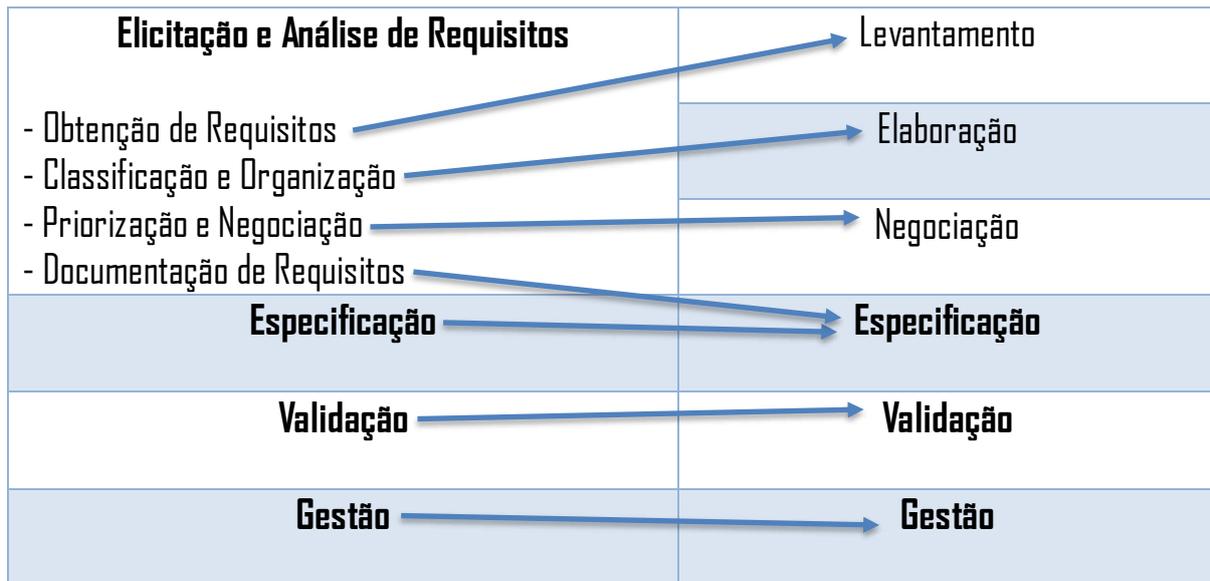
Gabarito: D

15. (FCC - 2011 - TRT - 19ª Região (AL) - Analista Judiciário - Tecnologia da Informação) De acordo com Sommerville, são atividades do processo de elicitación 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:





Conforme vimos em aula, trata-se de: Obtenção; Classificação e Organização; Priorização e Negociação; e Documentação.

Gabarito: E

16. (FCC - 2011 - INFRAERO - Analista de Sistemas - Desenvolvimento e Manutenção) 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:

Então, chegamos à atividade de priorização e negociação. Nessa fase, 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.

Galera... falou em Negociação, tem que lembrar de consenso entre as partes interessadas. E para chegar a um consenso, deve-se definir prioridades.

Gabarito: C

17. (FCC - 2011 - TRT - 1ª REGIÃO (RJ) - Analista Judiciário - Tecnologia da Informação) 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:

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.

Observação de Rotinas? Etnografia!

Gabarito: E

18. (FCC - 2009 - TRT - 7ª Região (CE) - Analista Judiciário - Tecnologia da Informação) 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:

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.

Já viram quantas vezes essa técnica caiu em provas? De novo, Etnografia!

Gabarito: E

19. (FCC - 2010 - MPE-RN - Analista de Tecnologia da Informação - Engenharia de Software) 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:



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.

Conforme vimos em aula, trata-se de uma técnica de observação para compreender requisitos.

Gabarito: C

20.(FCC - 2009 - TRT - 3ª Região (MG) - Analista Judiciário - Tecnologia da Informação) 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:

Pontos de Vista (Viewpoint-Oriented Requirements Definition – VORD):

Essa técnica 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.

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.*

Entrevistas:



*Entrevistas formais ou informais com os stakeholders no 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 das respostas a essas questões.***

Conforme vimos em aula, trata-se de: Pontos de Vista, Cenários e Entrevista.

Gabarito: D

21. (FCC - 2011 - INFRAERO - Analista de Sistemas - Desenvolvimento e Manutenção) 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 irreais (de desenvolvimento impossível).

Gabarito: E

22. (FCC - 2013 – DPE/SP – Analista de Sistemas) 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.

Comentários:

Essa atividade verifica 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.

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



23.(FCC - 2011 - TRT - 19ª Região (AL) - Analista Judiciário - Tecnologia da Informação) 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:

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.

Conforme vimos em aula, trata-se da Rastreabilidade, i.e., avaliar impacto no projeto da mudança em um requisito.

24.(FCC - 2009 - PGE-RJ - Técnico Superior de Análise de Sistemas e Métodos) 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:

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.

Conforme vimos em aula, a revisão técnica formal (revisão de requisitos) é uma técnica ou mecanismo de validação de requisitos.

Gabarito: E

25. (FCC - 2011 - TRT - 23ª REGIÃO (MT) - Analista Judiciário - Tecnologia da Informação) 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:

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.

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.

Gabarito: A

26. (FCC - 2010 - MPE-RN - Analista de Tecnologia da Informação - Engenharia de Software) 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:

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.

Conforme vimos em aula, trata-se do Gerenciamento de Requisitos.

Gabarito: E

27. (ESAF - 2005 – AFRFB - Analista de Sistemas - II) 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: C

28. (ESAF - 2005 – AFRFB - Analista de Sistemas - III) Durante a análise de requisitos, o principal foco do analista recai sobre como? e não sobre o que?. Nesse caso, o analista concentra-se em como o sistema produz ou consome dados, como o sistema deve executar as funções e como as restrições e interfaces são definidas.

Comentários:

Na verdade, a atividade de análise de requisitos se preocupa com O QUE? e a atividade de projeto se preocupa com COMO?.

Gabarito: E



29. (ESAF - 2005 – AFRFB - Analista de Sistemas - IV) 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: C

30. (ESAF - 2010 – CVM – Analista de Sistemas) 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: B



31. (ESAF - 2010 – MPOG – Analista de Sistemas) 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: B

32. (FGV - 2010 - BADESC - Analista de Sistemas - Desenvolvimento de Sistemas)
Analisar 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:

*Sommerville ainda diz que os requisitos não funcionais estão raramente associados às características individuais do sistema. Tratam-se de qualidades globais de um software, como manutenibilidade, usabilidade, desempenho, custos e várias outras. Normalmente estes requisitos são descritos de maneira informal. **Galera, em geral, afetam a arquitetura do sistema.***

Questão polêmica! No entanto, trata-se de um Requisito Não-Funcional!

Gabarito: E

33.(FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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: E

34.(FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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:

Requisitos Funcionais: são 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. Depende do tipo de software, dos usuários esperados e do tipo de sistema onde o software será implantado. Eles fazem parte da arquitetura do aplicativo de um sistema.

Conforme vimos em aula, são Requisitos Funcionais.

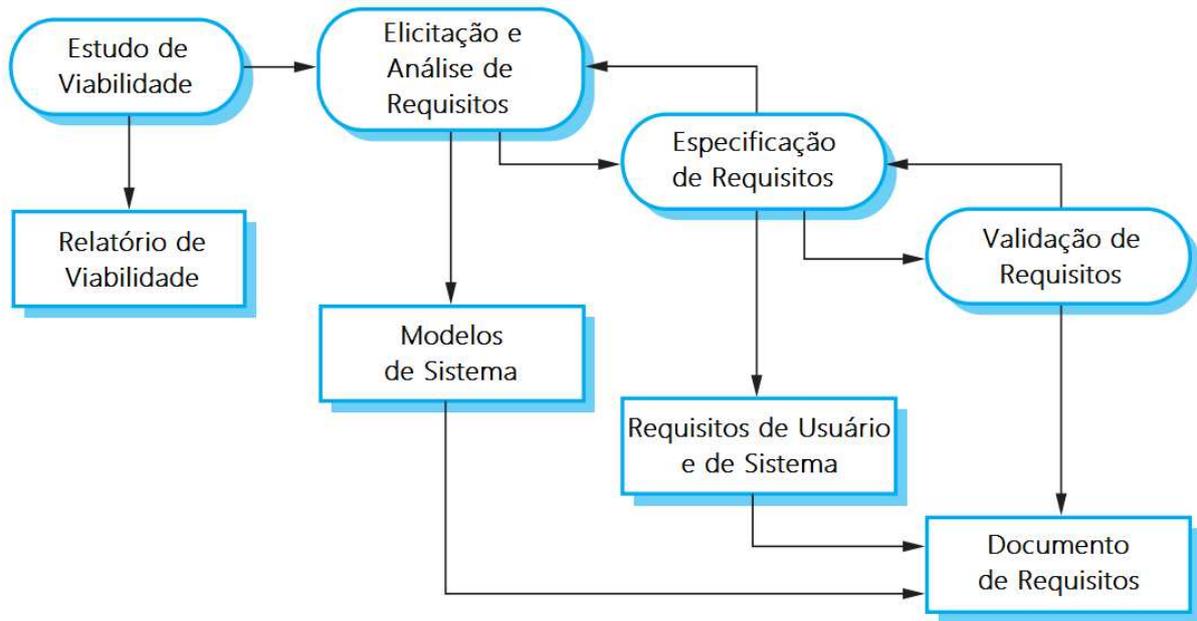
Gabarito: D

35. (FGV - 2008 - Senado Federal - Analista de Sistemas) 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: B

36. (FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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:

Cenários, Entrevistas, Protótipos, Workshop e Brainstorm!

Gabarito: C



37. (FGV - 2010 - FIOCRUZ - Tecnologista em Saúde - TI - Sistemas de Informação)

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:

É o *processo de levantamento e derivação de requisitos de sistema* através da observação de sistemas existentes, discussões com usuários potenciais e compradores, análise de tarefas, etc. 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.

Tranquilo, não? O objetivo principal é a obtenção dos requisitos!

Gabarito: A

38. (FGV - 2017 – ALERJ – Analista de Sistemas) 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: E

39. (CESGRANRIO – 2016 – IBGE – Supervisor de Pesquisas – Tecnologia de Informação e Comunicação) 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: C

40.(CESGRANRIO – 2014 – IBGE – Supervisor de Pesquisas - Tecnologia de Informação e Comunicação) 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: B

41. (CESGRANRIO – 2012 – CHESF – Profissional de Nível Superior) 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:

De acordo com Sommerville:

"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 de um conjunto preliminar de requisitos de negócio, um esboço da descrição do sistema e como o sistema pretende apoiar os processos de negócio. Os resultados do estudo de viabilidade devem estar em um relatório que recomenda se vale a pena ou não prosseguir com os processos de engenharia de requisitos e de desenvolvimento de sistema".

Logo, trata-se da Análise/Estudo de Viabilidade.

Gabarito: A

ACERTEI	ERREI



LISTA DE EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS) ENGENHARIA DE REQUISITOS

1. (FCC - 2012 - TST - Analista Judiciário - Análise de Sistemas) 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.
2. (FCC - 2012 - TCE-AP - Analista de Controle Externo - Tecnologia da Informação) 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.
3. (FCC - 2011 - INFRAERO - Analista de Sistemas - Gestão de TI – Letra D) No contexto de levantamento de requisitos, funcionalidade é um dos aspectos que deve ser levado em conta na abordagem dos requisitos funcionais.
4. (FCC - 2010 - BAHIA GÁS - Analista de Processos Organizacionais - Análise de Sistemas) 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.
5. (FCC - 2009 – SEFAZ/SP - Analista de Sistemas) *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.
6. (FCC - 2010 – DPE/SP - Analista de Sistemas) 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.
7. (FCC - 2009 - TRT - 3ª Região (MG) - Analista Judiciário - Tecnologia da Informação) 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.

8. (FCC - 2010 – DPE/SP - Analista de Sistemas - III) 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.

9. (FCC - 2013 – SEFAZ/SP – Analista de Sistemas) 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.
10. (FCC - 2013 – MPE/MA – Analista de Sistemas) 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.

11. (FCC - 2012 - MPE-PE - Analista Ministerial - Informática) 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.

12. (FCC - 2012 - TRE-CE - Analista Judiciário - Análise de Sistemas) 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.

13. (FCC - 2012 - TST - Analista Judiciário - Análise de Sistemas) 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.

14. (FCC - 2012 - TJ-PE - Analista Judiciário - Análise de Sistemas) 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.

15. (FCC - 2011 - TRT - 19ª Região (AL) - Analista Judiciário - Tecnologia da Informação) 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.

16. (FCC - 2011 - INFRAERO - Analista de Sistemas - Desenvolvimento e Manutenção) 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.

17. (FCC - 2011 - TRT - 1ª REGIÃO (RJ) - Analista Judiciário - Tecnologia da Informação) 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.

18. (FCC - 2009 - TRT - 7ª Região (CE) - Analista Judiciário - Tecnologia da Informação) 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.

19. (FCC - 2010 - MPE-RN - Analista de Tecnologia da Informação - Engenharia de Software) 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).

20. (FCC - 2009 - TRT - 3ª Região (MG) - Analista Judiciário - Tecnologia da Informação) 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.

21. (FCC - 2011 - INFRAERO - Analista de Sistemas - Desenvolvimento e Manutenção) 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.

22. (FCC - 2013 – DPE/SP – Analista de Sistemas) 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 - 2011 - TRT - 19ª Região (AL) - Analista Judiciário - Tecnologia da Informação) 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.

24. (FCC - 2009 - PGE-RJ - Técnico Superior de Análise de Sistemas e Métodos) 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.

25. (FCC - 2011 - TRT - 23ª REGIÃO (MT) - Analista Judiciário - Tecnologia da Informação) 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.

26. (FCC - 2010 - MPE-RN - Analista de Tecnologia da Informação - Engenharia de Software) 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.

27. (ESAF - 2005 – AFRFB - Analista de Sistemas - II) 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.

28. (ESAF - 2005 – AFRFB - Analista de Sistemas - III) Durante a análise de requisitos, o principal foco do analista recai sobre como? e não sobre o que?. Nesse caso,



o analista concentra-se em como o sistema produz ou consome dados, como o sistema deve executar as funções e como as restrições e interfaces são definidas.

29. (ESAF - 2005 – AFRFB - Analista de Sistemas - IV) 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.

30. (ESAF - 2010 – CVM – Analista de Sistemas) 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.

31. (ESAF - 2010 – MPOG – Analista de Sistemas) 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.

32. (FGV - 2010 - BADESC - Analista de Sistemas - Desenvolvimento de Sistemas)
Analisar 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.

33. (FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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.

34. (FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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.

35. (FGV - 2008 - Senado Federal - Analista de Sistemas) 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.

36. (FGV - 2009 - MEC - Analista de Sistemas - Especialista) 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.

37. (FGV - 2010 - FIOCRUZ - Tecnologista em Saúde - TI - Sistemas de Informação) 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.

38. (FGV - 2017 - ALERJ - Analista de Sistemas) 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.

39. (CESGRANRIO – 2016 – IBGE – Supervisor de Pesquisas – Tecnologia de Informação e Comunicação) 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.

40. (CESGRANRIO – 2014 – IBGE – Supervisor de Pesquisas - Tecnologia de Informação e Comunicação) 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



41. (CESGRANRIO – 2012 – Chesf – Profissional de Nível Superior) 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



GABARITO DOS EXERCÍCIOS COMENTADOS (DIVERSAS BANCAS)

ENGENHARIA DE REQUISITOS

1	2	3	4	5	6	7	8	9	10
C	A	C	A	A	C	E	A	C	B
11	12	13	14	15	16	17	18	19	20
A	C	C	D	E	C	E	E	C	D
21	22	23	24	25	26	27	28	29	30
E	E	A	E	A	E	C	E	C	B
31	32	33	34	35	36	37	38	39	40
B	E	E	D	B	C	A	E	C	B
41	42	43	44	45	46	47	48	49	50
E									



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.