

**Aula 00— Conceitos básicos
sobre Engenharia de Software,
~~Disciplinas de Engenharia de~~
~~Software, Análise de Requisitos~~**

Sumário

SUMÁRIO	2
O QUE ESTUDAREMOS NA AULA DE HOJE?	4
INTRODUÇÃO	4
CONCEITOS BÁSICOS.....	4
DO HISTÓRICO.....	5
DEFINIÇÃO DE SOFTWARE	5
CAMPOS DE APLICAÇÃO	6
DISCIPLINAS DA ENGENHARIA DE SOFTWARE	8
ANÁLISE DE REQUISITOS	10
CONCEITO.....	13
ANÁLISE DE REQUISITOS	13
REQUISITOS	15
VALIDAÇÃO DE REQUISITOS	15
ESPECIFICAÇÃO DE REQUISITOS	16
CENÁRIOS DE USO.....	16
ARTEFATOS DO LEVANTAMENTO DE REQUISITOS	17
REQUISITOS FUNCIONAIS	18
REQUISITOS NÃO FUNCIONAIS.....	19
REQUISITOS DE USUÁRIOS DE SISTEMA.....	20
TIPOS DE ESPECIFICAÇÃO DE REQUISITOS	21
MÉTRICAS PARA ESPECIFICAR REQUISITOS NÃO FUNCIONAIS	21
USUÁRIOS DO PROCESSO	22
ESTRUTURA DE DOCUMENTO DE REQUISITOS	22
QUESTÕES COMENTADAS PELO PROFESSOR.....	23
LISTA DE QUESTÕES.....	52
GABARITO	73
RESUMO DIRECIONADO	74
CONCEITOS BÁSICOS.....	74
DO HISTÓRICO.....	74
DEFINIÇÃO DE SOFTWARE	75
CAMPOS DE APLICAÇÃO	75
DISCIPLINAS DA ENGENHARIA DE SOFTWARE	77

Neste material você terá:

Curso completo em VÍDEO

teoria e exercícios resolvidos sobre TODOS os pontos do edital (Conceitos básicos sobre Engenharia de software. Disciplinas de Engenharia de Software. Análise de Requisitos Funcionais e não Funcionais).

Curso completo escrito (PDF)

teoria e MAIS exercícios resolvidos sobre TODOS os pontos do edital (Conceitos básicos sobre Engenharia de software. Disciplinas de Engenharia de Software. Análise de Requisitos Funcionais e não Funcionais).

Acesso direto ao professor

para você sanar suas dúvidas DIRETAMENTE conosco sempre que precisar

Você nunca estudou Conceitos básicos sobre Engenharia de Software. Disciplinas da Engenharia de Software. Análise de Requisitos Funcionais e não Funcionais . Para concursos?

Se a sua resposta foi: "Não, nunca estudei!". Não tem problema, porque esse curso também é para você.

Nós abordaremos a teoria necessária para compreensão e resolução dos exercícios, no intuito de fazer você praticar e aprender exaustivamente cada assunto estudado para se sentir confiante a enfrentar a Banca Examinadora (Cespe/Cebraspe ou outra Banca que venha ser a escolhida).

Ah! Fique à vontade para me procurar no fórum de dúvidas sempre que for necessário. Estamos sempre à disposição.

Caso queira tirar alguma dúvida, basta enviar-me um e-mail ou um direct pelo Instagram.



O que estudaremos na aula de hoje?

Conceitos básicos sobre Engenharia de software. Disciplinas de Engenharia de Software. Análise de Requisitos Funcionais e não Funcionais.

Introdução

Meu caro aluno e aluna, falaremos sobre a engenharia de software e seus diversos aspectos, mas Prof. Para que serve essa tal engenharia de software, quais seus fundamentos, quais suas vertentes, e o que eu irei fazer com mais esta disciplina? Calma, responderemos todas as suas indagações, meu querido e minha querida Padawan!

Pessoalmente, gosto muito da obra de Pressman e de sua abordagem nas definições e conceitos do que realmente é a engenharia de software, entretanto temos outras enciclopédias referência no assunto, faremos uma miscigenação das duas principais obras na área da Engenharia de Software, também, com base no que cai em suas questões de provas da área, acompanhadas de exercícios sanando assim todas as dúvidas dentro do assunto ministrado.

Conceitos básicos

Dos conceitos da Engenharia de Softwares, temos:

“Engenharia de software é o estabelecimento e o emprego de sólidos princípios de engenharia de modo a obter software de maneira econômica, que seja confiável e funcione de forma eficiente em máquinas reais”.

“Engenharia de *software* é uma área da computação voltada à especificação, desenvolvimento, manutenção e criação de *software*, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando a organização, produtividade e qualidade”.

Pessoal, com o fundamento nos dois conceitos descritos acima, sobre o que vem a ser a Engenharia de Software, já conseguiremos desenhar um pouco do que é esta disciplina e seus fundamentos e como são empregados em nosso dia a dia. Atualmente, as tecnologias e práticas da Engenharia de Software, englobam as mais diversas linguagens de programação, banco de dados, ferramentas e entre elas as IDEs, plataformas, bibliotecas, padrões de projeto de software, processo de software e qualidade de software. Entretanto, a engenharia de software deve oferecer mecanismos para que possamos planejar e gerenciar o processo de desenvolvimento de um sistema computacional de qualidade e que atenda às necessidades de um requisitante de software.

As bases científicas para a Engenharia de Software se relacionam com o uso de modelos abstratos que permitem ao engenheiro especificar, projetar, implementar e manter sistemas de software, avaliando e garantindo suas qualidades. A área responsável que estuda e avalia os processos da engenharia de software, propondo a evolução dos processos, ferramentas e métodos de suporte a engenharia.

Do Histórico

O termo Engenharia de Software foi criado na década de 1960, porém só utilizado oficialmente por volta do ano 1968, na NATO Science Commite. Sua criação, emergiu na tentativa de contornar a crise do software e dar um tratamento de engenharia mais sistemático, controlado e qualidade mensurável ao desenvolvimento de sistemas e software complexos.

Um sistema de software complexo caracteriza-se por um conjunto de componentes abstratos de software, estruturas de dados e algoritmos encapsulados na forma de algoritmos, funções, módulos, objetos ou agentes interconectados, compondo a arquitetura do software, que deverão ser executados em sistemas computacionais.

No início, e ainda hoje em algumas empresas menores com a falta de profissionais experientes, desenvolver um sistema é uma jornada quase que infinita, ao oposto disso, a Engenharia veio para: criar modelos, padrões, processos, metodologias, documentações, uma linha de desenvolvimento. Pois como narrei no período inicial deste parágrafo, este processo era uma aventura e que na maioria das vezes a empresa a qual pagava pela construção de um determinado software sempre ficava no prejuízo e as razões eram as mais diversas, podemos até citar algumas destas: tempo prolongado no desenvolvimento, linguagem de programação não adequada, a não realização de levantamentos de requisitos funcionais e não funcionais, antes de entrar em produção o software já não atendia aos requisitos básicos ao qual foi construído, entre centenas de problemas que poderiam causar o seu insucesso.

A figura abaixo nos dá uma definição básica de fases da construção de um software.



Quando usamos das ferramentas certas ou adequadas, os métodos eficientes, o processo documentado e seguindo um fluxo, a qualidade se torna o resumo desta abstração.

Definição de Software

Amigos, começamos falando da Engenharia de Software, mas o que é realmente um software?

“É uma sequência de instruções a ser seguidas e/ou executadas, na manipulação, redirecionamento ou modificação de um dado (informação) ou acontecimento”.

“Software, também é o nome dado ao comportamento exibido por essa sequência de instruções, quando executada em um computador ou máquina semelhante, além de um produto desenvolvido pela engenharia de software, e inclui não só o programa de computador propriamente dito, mas também manuais e especificações”.

A respeito do software, para fins contábeis e financeiros, ele é considerado um bem de consumo!

“Um software normalmente é composto por diversas funções, bibliotecas e módulos que gera um programa executável ao final do processo de desenvolvimento e este, quando executado, recebe algum tipo de entrada de dados, no caso (*input*), processa as informações segundo uma série de algoritmos ou sequências de instruções lógicas e libera uma saída, esta, o (*output*) como resultado deste processamento. Um software bem desenvolvido é normalmente criado pela área da engenharia de software e inclui não apenas o programa de computador em si, mas também manuais, especificações e configurações”.

Bom pessoal, agora que sabemos na prática o que é um software, vamos partir para suas aplicações.

Campos de Aplicação

Os campos de aplicações são os mais diversos, pois a Engenharia de Software surgiu para colocar ordem na casa, rs. Antigamente, se desenvolvia um projeto de qualquer maneira, as datas de entrega da ferramenta quase sempre eram desobedecidas, construía-se algo de qualquer jeito, o que, para o cliente no final, não era o resultado desejado, e isto, não ocorria somente para software de sistemas, mas para diversos modelos de negócios, segue a relação das principais áreas em que a Engenharia de Software é aplicada:

- **Software de sistema:** estes são, conjuntos de programas desenvolvidos para atender a outros programas ou necessidades fins. A exemplo temos: compiladores, editores e utilitários para gerenciamento de arquivos que processam estruturas de informações complexas, outras aplicações de sistema, por exemplo, componentes de sistema operacional, drivers, softwares de redes de computadores, processadores de telecomunicações que processam dados amplamente indeterminados. Em ambos os casos, a área de software de sistemas é caracterizada por uma enorme interação com o hardware dos computadores, o uso intenso por múltiplos usuários, as operações concorrentes que requerem escalas de ordens altíssimas, compartilhamento de recursos e gestão de processo sofisticados; estruturas de dados complexas e múltiplas interfaces externas.
- **Software de aplicação:** são programas específicos que solucionam uma necessidade específica de negócio. Aplicados nesta área, processam dados comerciais ou técnicos de uma forma que facilite as operações comerciais ou tomadas de decisões administrativas/técnicas. Além das

aplicações convencionais de processamento de dados, o software de aplicação é usado para controlar funções de negócio em tempo real, a exemplo, processamento de transações em pontos de venda, controle de processos de fabricação em tempo assíncrono.

- **Software científico / engenharia:** de uso e característico por algoritmos **number crunching**, para o processamento numérico pesado. As aplicações neste modelo vão da astronomia à vulcanologia, da análise de tensões na indústria automotiva à dinâmica orbital de ônibus espaciais, também da biologia molecular à fabricação automatizada. Entretanto, aplicações modernas dentro da área de engenharia/científica estão se afastando dos algoritmos numéricos convencionais. Projeto com o auxílio de computador, simulação de sistemas e outras aplicações interativas começaram a ter características de sistemas em tempo real e até mesmo de software de sistemas.
- **Software embutido:** residente em produtos específicos ou sistema é utilizado para implementar e controlar características e funções para o usuário final e para o próprio sistema. Executa funções limitadas e específicas, a exemplo: controle do painel de um forno micro-ondas ou fornece funções significativas e capacidade de controle, funções digitais de automóveis, tal como controle do nível de combustível, painéis de controle e sistemas de freios.
- **Software para linha de produtos:** projetado para prover a capacidade específica de utilização por muitos clientes diferentes. Pode focalizar um mercado limitado e particularizado: produtos para controle de estoques ou direcionar-se para os mercados de consumo de massa, processamento de texto, planilhas eletrônicas, computação gráfica, multimídia, entretenimento, gerenciamento de bancos de dados e aplicações financeiras pessoais e comerciais.
- **Aplicações para a Web:** chamadas de WebApps, essa categoria de software centralizada em redes, alcança uma vasta gama de aplicações. Em sua forma mais simples, as WebApps podem ser pouco mais que um conjunto de arquivos de hipertexto interconectados, apresentando informações por meio de texto e gráficos limitados. Entretanto, com o aparecimento da Web 2.0, elas têm evoluído para sofisticados ambientes computacionais que não apenas fornecem recursos especializados, mas funções computacionais e conteúdo para o usuário final, como a integração de bancos de dados corporativos e aplicações comerciais.
- **Software de inteligência artificial:** faz uso de algoritmos não numéricos para solucionar problemas complexos que não são passíveis de computação ou de análise direta. Aplicações nessa área incluem: robótica, sistemas especialistas, reconhecimento de padrões de imagem e voz, redes neurais artificiais, prova de teoremas e jogos. Milhões de engenheiros de software em todo o mundo trabalham arduamente em projetos de software nas mais diversas categorias. Em alguns casos, novos sistemas estão sendo construídos, porém, em outros casos as aplicações já existentes e estão sendo corrigidas, adaptadas e aperfeiçoadas. Não é incomum para um jovem engenheiro de software trabalhar em um programa mais velho que ele.

- **Computação mundial aberta:** o acelerado crescimento das redes sem fio pode, em breve, conduzir a uma verdadeira computação distribuída e ampliada, compartilhada e incorporada nos ambientes domésticos e comerciais. O desafio para os engenheiros de software será o de desenvolver sistemas e software aplicativo que permitam que dispositivos móveis, computadores pessoais e sistemas corporativos se comuniquem através de extensas redes. O que hoje já acontece em grande escala!
- **Netsourcing (recursos via Internet):** a Internet rapidamente tornou-se um grande e avassalador mecanismo computacional, como um provedor de conteúdo. O desafio para os engenheiros de software, consiste em arquitetar aplicações simples, isto é, planejamento financeiro, pessoal e sofisticados que forneçam benefícios aos mercados mundiais de usuários finais visados.
- **Software aberto:** tendência crescente que resulta na distribuição de código-fonte para aplicações de sistemas, por exemplo: sistemas operacionais, bancos de dados e ambientes de desenvolvimento, de forma que muitas pessoas possam contribuir para seu crescimento. O desafio para os engenheiros de software, consiste em construir um código fonte auto descritivo, mais importante ainda é, desenvolver técnicas que permitam que tanto os clientes quanto os desenvolvedores saibam quais alterações foram feitas e como se manifestam dentro do software.

Disciplinas da Engenharia de Software

Dentro do processo da Engenharia de Software, iremos destacar as principais disciplinas, as quais despencam em provas de concursos.:

1. **Requisitos de software:** Na sistematização e engenharia de software, **análise de requisitos** engloba todas as tarefas que lidam com a investigação, definição e escopo de novos sistemas ou alterações. Análise de requisitos é parte importante do processo de desenvolvimento de softwares, na qual o engenheiro de requisitos e o analista de negócio, juntamente com engenheiros de sistema ou desenvolvedores de software, identificam as necessidades ou requisitos de um cliente. Uma vez que os requisitos do sistema tenham sido identificados, os projetistas de sistemas estarão preparados para projetar a solução.
A análise de requisitos é uma das primeiras atividades no desenvolvimento de software. O produto do seu trabalho é a especificação dos requisitos, esta, define o escopo do software em duas dimensões: **Requisito funcional e Requisito não-funcional**. É nesta fase que o analista faz as primeiras reuniões com os clientes e/ou usuários do software para conhecer as funcionalidades do sistema que será desenvolvido. Também, ocorre a maior parte dos erros, pois a falta de experiência dos clientes ou usuários faz com que nem sempre tenham claro em sua mente quais funcionalidades o software terá.

Outro conceito é: *Análise de Requisitos é o processo que envolve o estudo das necessidades do usuário para encontrar-se uma definição correta ou completa do sistema ou requisito de software.*

2. **Projeto de software:** parte responsável da engenharia de software que se encarrega de fazer todo o planejamento anterior ao desenvolvimento, incluindo a definição da arquitetura do software, e transformar tudo em um documento ou conjunto de documentos capazes de ser interpretados diretamente pelo programador. Para atingir este objetivo, o projetista deve mapear as estruturas e funcionalidades identificadas na análise de requerimentos dentro do contexto e das restrições da arquitetura, de forma a tornar possível a construção do software. Ao longo do tempo e nos diversos processos de software existentes, várias ferramentas foram idealizadas para facilitar e atingir este objetivo.
3. **Construção de software:** Um processo de desenvolvimento de software é um conjunto de atividades, parcialmente ordenadas, com a finalidade de obter um produto de software. Estudado dentro da área de Engenharia de Software, considerado um dos principais mecanismos para se obter software de qualidade e cumprir corretamente os contratos de desenvolvimento, sendo uma das respostas técnicas adequadas para resolver a crise do software. Este é um dos pilares da entrega de produto em tempo hábil.
4. **Teste de software:** O teste do software é a investigação do funcionamento do mesmo, a fim de fornecer informações sobre sua qualidade em relação ao contexto em que ele deve operar, se relaciona com o conceito de verificação e validação. Estas que foram levantadas através de requisitos funcionais e não funcionais. Isto inclui o processo de utilizar o produto para encontrar seus defeitos. O teste é um processo realizado pelo testador de software, que permeia outros processos da engenharia, e que envolve ações que vão do levantamento de requisitos até a execução do teste propriamente dito.
5. **Manutenção de software:** Em engenharia de software, manutenção de software é o processo de melhoria e otimização de um software já desenvolvido, lembrando que, um produto mal desenvolvido com certeza necessitará de manutenções constantes e reparos de defeitos. A manutenção do software é uma das fases do processo de desenvolvimento de software, ocorre após a entrada do software em produção. A manutenção pode ser preventiva ou mesmo corretiva. Também ocorre quando aponta para falhas no Processo de Desenvolvimento de Software - PDS que devem ser identificadas e tratadas, visando a melhoria contínua do processo. As anomalias são inseridas pelo processo em si ou pelo não entendimento dos requisitos do produto.
6. **Gerência de configuração de software:** A Gerência de configuração de software ou gestão de configuração de software é, a área da engenharia de software responsável por fornecer o apoio para o desenvolvimento deste. Suas atribuições básicas são o controle de versão, o controle de mudança e a auditoria das configurações.
7. **Gerência de engenharia de software:** A Engenharia de software é, a área da computação voltada à especificação, desenvolvimento, manutenção e criação de software, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando organização, produtividade e qualidade. Atualmente, essas tecnologias e práticas englobam linguagens de programação, banco de dados, ferramentas, plataformas, bibliotecas, padrões de projeto de software, processo de software e qualidade de software. Além disso, a engenharia de software deve oferecer mecanismos para se planejar e gerenciar o processo de desenvolvimento de um sistema computacional de qualidade e que atenda às necessidades de um requisitante de software.
8. **Processos de Engenharia de Software:** Um processo de Engenharia de Software é composto por um conjunto de passos de processo parcialmente ordenados, relacionados com artefatos, pessoas,

recursos, estruturas organizacionais e restrições, tendo como objetivo produzir e manter os produtos de software finais requeridos.

9. **Ferramentas e Métodos de Engenharia de Software:** Outro ponto crucial é o uso de ferramentas CASE do inglês Computer-Aided Software Engineering. Essa classificação abrange toda ferramenta baseada em computadores que auxiliam as atividades da engenharia de software, desde a análise de requisitos, modelagem e até fase de testes. Os ambientes de desenvolvimento integrado (IDEs) têm maior destaque e suportam, entre outras:

- ✓ Editor.
- ✓ Debug.
- ✓ Geração de código.
- ✓ Modelagem.
- ✓ Deploy.
- ✓ Testes não automatizados.
- ✓ Testes automatizados.
- ✓ Refatoração (Refactoring).
- ✓ Gestão de Riscos nos projetos de Software.
- ✓ Uso da Prototipagem na Eng. de Requisitos.

10. **Qualidade de software:** O significado de qualidade possui várias definições na literatura. O glossário do IEEE define qualidade como Grau de conformidade de um sistema, componente ou processo com os respectivos requisitos ou alternativamente, como Grau de conformidade de um sistema, componente ou processo com as necessidades e expectativas de clientes ou usuários.

Ambas as definições refletem os aspectos importantes da qualidade, diversos autores apresentam outras definições que, geralmente giram em torno dos temas de conformidade com os requisitos e atendimento das expectativas. Naturalmente, pode haver diferenças entre as aplicações dessas definições se os requisitos explícitos não refletirem corretamente as necessidades reais.

Análise de Requisitos

Levantamento de Requisitos É a disciplina parte da Engenharia de Software que engloba todas as tarefas no que se refere, a investigação, definição e escopo de novos sistemas, ou mesmo, alterações destes. Análise de requisitos é algo primordial no processo de desenvolvimento de um software.

Nesta disciplina, o engenheiro de requisitos e o analista de negócio, juntamente com o engenheiro de sistema, identificam com o uso de técnicas, as necessidades ou requisitos de um determinado cliente. O processo da análise e requisitos é realizado através de entrevistas com o cliente, este por sua vez, beneficiado pelo software que será desenvolvido. Um requisito é a capacidade pela qual um resultado do projeto (produto ou serviço) deve obedecer.

A rastreabilidade é estabelecida e usada no levantamento de requisitos, ela informa o compromisso da companhia e os interesses das partes em termos de conformidade, integralidade, cobertura e consistência.

A análise de requisitos envolve a comunicação entre os membros da equipe do projeto e as partes interessadas, além da adequação às mudanças de requisitos ao longo do curso do projeto. Para evitar que uma classe de requisitos substitua outra a comunicação constante entre os membros da equipe de desenvolvimento é fundamental.

O que é levantamento de requisitos? Abordaremos de maneira simplificada para que caiba no entendimento de todos, dessa maneira **levantamento de requisitos** é o processo de compreensão e identificação

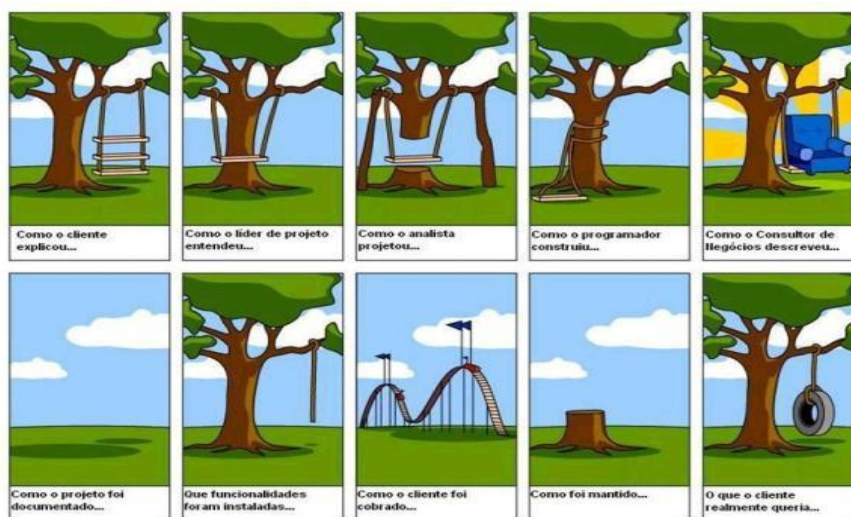
das necessidades do cliente, que espera a sua resolução da demanda, a ser solucionada pelo sistema ou aplicativo que será desenvolvido e bem definido em suas funções, ou seja, o que este software irá realizar.

Algo árduo neste processo é, a comunicação entre a equipe de entrevista com o cliente e o próprio, essa realizará o levantamento das reais necessidades da empresa, também, estará a par do que o cliente acha que é a sua necessidade, rs... lembro em que certa vez, trabalhei em uma empresa na qual fiquei responsável por realizar o levantamento de requisitos do setor financeiro, ao sentar ao lado do responsável daquele setor, notei que a pessoa sabia sobre tudo a respeito das rotinas do financeiro, como funcionava cada detalhe das regra de negócio, mas claramente percebi que ele não sabia passar realmente as suas necessidades.

Às vezes é complexo e frustrante, pois nos traz uma sobrecarga de trabalho, afinal de contas, caso algo saia errado ou você esqueça de algum detalhe, algo que seja importante e que passou despercebido no que se refere ao desenvolvimento do sistema, o problema será seu e não do cliente.

Outra situação complicada neste processo é, durante as entrevistas, momento em que realizamos o levantamento de requisitos: no que se refere às necessidades daquele setor, você anota tudo e com riqueza de detalhes descreve também todo o processo e as regras de negócios do estabelecimento ou departamento, mais à frente, na apresentação do sistema para o cliente, caso algo dê errado, seja por entendimento errôneo seu ou pela maneira como o cliente passou e que não condiz com o fluxo de trabalho, já vou te avisando logo com todo o carinho do mundo, isto acontece muito. Algo que aprendi nesta jornada foi gravar todo o processo de entrevista com o cliente, pois é chato desenvolver algo que não será usado e que o cliente vai afirmar que não te passou da maneira que foi construído no sistema, mas acredite, ele falou sim! Rs....

Acredito que todos que estão lendo este material conhecem esta tradicional figurinha! Ela define como na maioria das vezes funciona a coisa!



(CESPE - 2018 - Polícia Federal - Agente de Polícia Federal).

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

() Certo.

() Errado.

RESOLUÇÃO:

Exatamente, na fase de levantamento de requisitos ou análise de requisitos, é a fase em que realizamos todos os levantamentos, regras de negócio do cliente, necessidade real do cliente e de posse destas informações, podemos definir, banco de dados que será utilizado no sistema e a linguagem de programação.

Resposta: Certo

(CESPE - 2015 - CGE-PI - Auditor Governamental - Tecnologia da Informação).

Acerca de gerenciamento de requisitos, julgue o item que se segue.

As atividades do processo de engenharia de requisitos abrangem o estudo de viabilidade do sistema; a obtenção e a análise de requisitos; a especificação e a validação de requisitos; e finalmente, a aprovação dos requisitos pelos clientes.

() Certo.

() Errado.

RESOLUÇÃO:

No processo de validação de requisitos, não é o cliente que aprova os requisitos, mais sim, o analista e sua equipe.

Resposta: Errado

(CESPE - 2018 - IPHAN - Analista I - Área 7).

No que se refere à engenharia de requisitos, julgue o próximo item.

Situação hipotética: O fluxo de processos para o gerenciamento de requisitos de uma organização contém, entre outras, as seguintes atividades: estudo de viabilidade; elicitação de requisitos; e análise de requisitos.

Assertiva: A elicitação de requisitos deve ser retirada do fluxo, porque não faz parte do gerenciamento de requisitos, devendo ser executada antes da realização desse gerenciamento.

() Certo.

() Errado.

RESOLUÇÃO:

No processo de validação de requisitos, todas estas tarefas, fazem parte do processo de análise, portanto, não serão dispensadas de maneira alguma.

Resposta: Errado

Conceito

Pessoal, sei que já falei na introdução acima o conceito de Requisitos de Software, mas quero enfatizar mais uma vez, de maneira bem técnica, então vamos lá?

Na sistematização e engenharia de software, **análise e técnicas de levantamento de requisitos** engloba todas as tarefas que lidam com investigação, definição e escopo de novos sistemas ou alterações. Análise de requisitos é parte importante do processo de desenvolvimento de softwares, na qual o engenheiro de requisitos e o analista de negócios juntamente com o engenheiro de sistema ou desenvolvedor de software identificam as necessidades ou requisitos de um cliente. Uma vez que os requisitos do sistema tenham sido identificados os projetistas de sistemas estarão aptos a projetar a solução.

Análise de Requisitos

A análise de requisitos é uma das primeiras atividades de desenvolvimento de software. O produto do seu trabalho é a especificação de requisitos, que define o escopo do software em duas dimensões: **Requisito funcional e Requisito não-funcional**. É nesta fase que o analista realiza as primeiras reuniões com o cliente e/ou usuários do software para conhecer as funcionalidades do sistema que será desenvolvido. Também é nesta fase que ocorrem a maior parte dos erros, pois a falta de experiência do cliente ou usuários faz com que eles nem sempre tenham claro em sua mente quais funcionalidades o software portará.

As entrevistas estruturadas são um método utilizado nesta fase e que obterão um papel importante na ajuda à compreensão de todas as funcionalidades pretendidas pelo cliente.

A análise de requisitos é conhecida ainda por outros nomes, são eles: **Engenharia de Requisitos, Levantamentos de Requisitos, Captura de Requisitos, Análise de sistemas, Especificação de Requisitos e Análise de Requerimentos**.

Em meu entendimento uma das melhores obras, se não a melhor, referente a este assunto é a de Roger Pressman, pois detalha com riqueza o processo entre outros dentro da disciplina Engenharia de Software.

Em sua obra, ele destaca a importância de identificarmos os **interessados**. Nesse sentido, os interessados são qualquer um que se beneficia de forma direta ou indireta do sistema que está sendo desenvolvido – usuários, gerente de operações, gerente de produtos, engenheiro de produtos, engenheiro de softwares, usuários finais consultores internos. Assim, pessoas dos mais diversos setores da empresa com visões diferentes são entrevistadas pelos analistas levantadores de requisitos para que a análise possa ser completa e atinja o fim que almeja.

Outro conceito é: *Análise de Requisitos é o processo que envolve o estudo das necessidades do usuário para encontrar uma definição correta ou completa do sistema ou requisito de software.*

Vocês observaram que de diferentes formas tenho narrado sempre o conceito inicial do que é Análise de Requisitos? Isto mesmo, quero que vocês fixem este termo, pois só com isso bem claro em mente vocês conseguirão resolver diversas questões de provas, mostrarei isto mais à frente.

A comunicação é fundamental neste processo, uma vez que traz o cliente para perto de modo que eles influenciem também na entrega e no sucesso do produto.

A Análise e Técnicas de Levantamento de Requisitos consiste em:

- **Reconhecer o Problema:** Concentra-se nas especificações do sistema, todo o planejamento, o contato do profissional de TI, no caso o Analista de Requisitos com o cliente. A intenção é a de entender a real necessidade do cliente, abstraídas no processo de entrevista com o próprio.
- **Avaliar o Problema e a Síntese da Solução:** tem-se o entendimento do problema, de posse do conhecimento da real situação do cliente e do que este almeja com o software, apontaremos a melhor solução para ele.
- **Modelagem:** este recurso é utilizado para o suporte da solução. Como assim professor? Pessoal, o modelo ou a modelagem, vai apresentar o sistema ou as telas que auxiliarão no entendimento e compreensão do sistema com as funcionalidades do programa que está sendo desenvolvido.
- **Especificação de Requisitos:** esta fase é responsável por consolidar, funções, interfaces, desempenho, no contexto de restrições do sistema desenvolvido.
- **Revisão:** neste período do processo, juntos, cliente e analistas, avaliarão o real objetivo do projeto com o a finalidade de desfazer-se de possíveis redundâncias existentes, inconsistências entre outros.

(CESPE - 2018 - MPE-PI - Analista Ministerial - Tecnologia da Informação).

A respeito da engenharia de *software*, julgue o seguinte item.

A análise de requisitos consiste na área responsável pela identificação das reais necessidades dos clientes de TI. Por meio da análise de requisitos, em conjunto com o cliente, é possível construir uma solução que atenda essas necessidades e desenvolver os requisitos funcionais elencados.

() Certo.

() Errado.

RESOLUÇÃO:

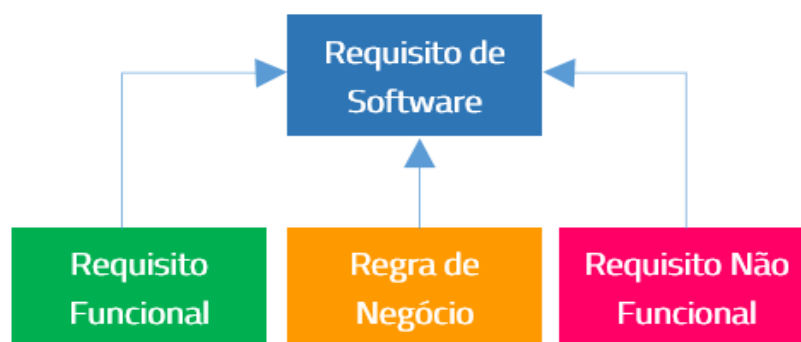
Acho que este conceito de análise de requisito já está bem fixado na cabeça de vocês! Rs.... É isso mesmo, por meio da análise de requisitos, podemos chegar a uma solução que satisfará as necessidades do cliente no que tange a um produto com excelência.

Resposta: Certo

Requisitos

No âmbito da Engenharia de Software, um **Requisito** consiste na definição documentada de uma propriedade ou comportamento que um produto ou serviço em particular deve atender.

Na abordagem clássica da engenharia conjuntos de requisitos são tipicamente utilizados como informações fundamentais para a fase de projeto de um produto ou serviço, especificando as propriedades e funções fundamentais para a fase de projeto de um produto ou serviço.



Na imagem abaixo, temos um exemplo clássico de uma reunião para levantarmos os primeiros requisitos.



Condução de uma reunião para levantamento de requisitos

Cena: Uma sala de reunião. A primeira reunião para levantamento de requisitos está em andamento.

Atores: Jamie Lazar, membro da equipe de software; Vinod Raman, membro da equipe de software; Ed Robbins, membro da equipe de software; Doug Miller, gerente da engenharia de software; três membros do Depto. de Marketing; um representante da Engenharia de Produto e um facilitador.

Conversa:

Facilitador (apontando para uma lousa branca): Portanto, esta é a lista atual de objetos e serviços para a função segurança domiciliar.

Representante do Depto. de Marketing:

Segundo o nosso ponto de vista, esta lista cobre aproximadamente todas as funcionalidades.

Vinod: Alguém não mencionou que eles queriam que toda a funcionalidade do CasaSegura pudesse ser acessada via Internet?

Isso incluiria a função de segurança domiciliar, não é mesmo?

Representante do Depto. de Marketing: Sim, você está certo... Teremos de acrescentar essa funcionalidade e os objetos apropriados.

Facilitador: Isso também não acrescentaria algumas restrições?

Jamie: Realmente, restrições técnicas e legais.

Representante da Engenharia de Produto: E isso significa o quê?

Jamie: É melhor termos certeza de que um intruso não conseguirá entrar no sistema, desarmá-lo e roubar o local ou coisa pior. Grande responsabilidade sobre nós.

Doug: Uma grande verdade.

Representante do Depto. de Marketing: Mas ainda assim precisamos que... Apenas ter certeza de impedir que um intruso entre.

Ed: É fácil dizer, o duro é fazer...

Facilitador (interrompendo): Não gostaria de discutir esta questão agora. Anotemos a questão para ser trabalhada no futuro e prossigamos.

(Doug, atuando como o secretário da reunião, faz um apontamento apropriado.)

Facilitador: Tenho a impressão de que ainda há mais coisas a ser consideradas aqui.

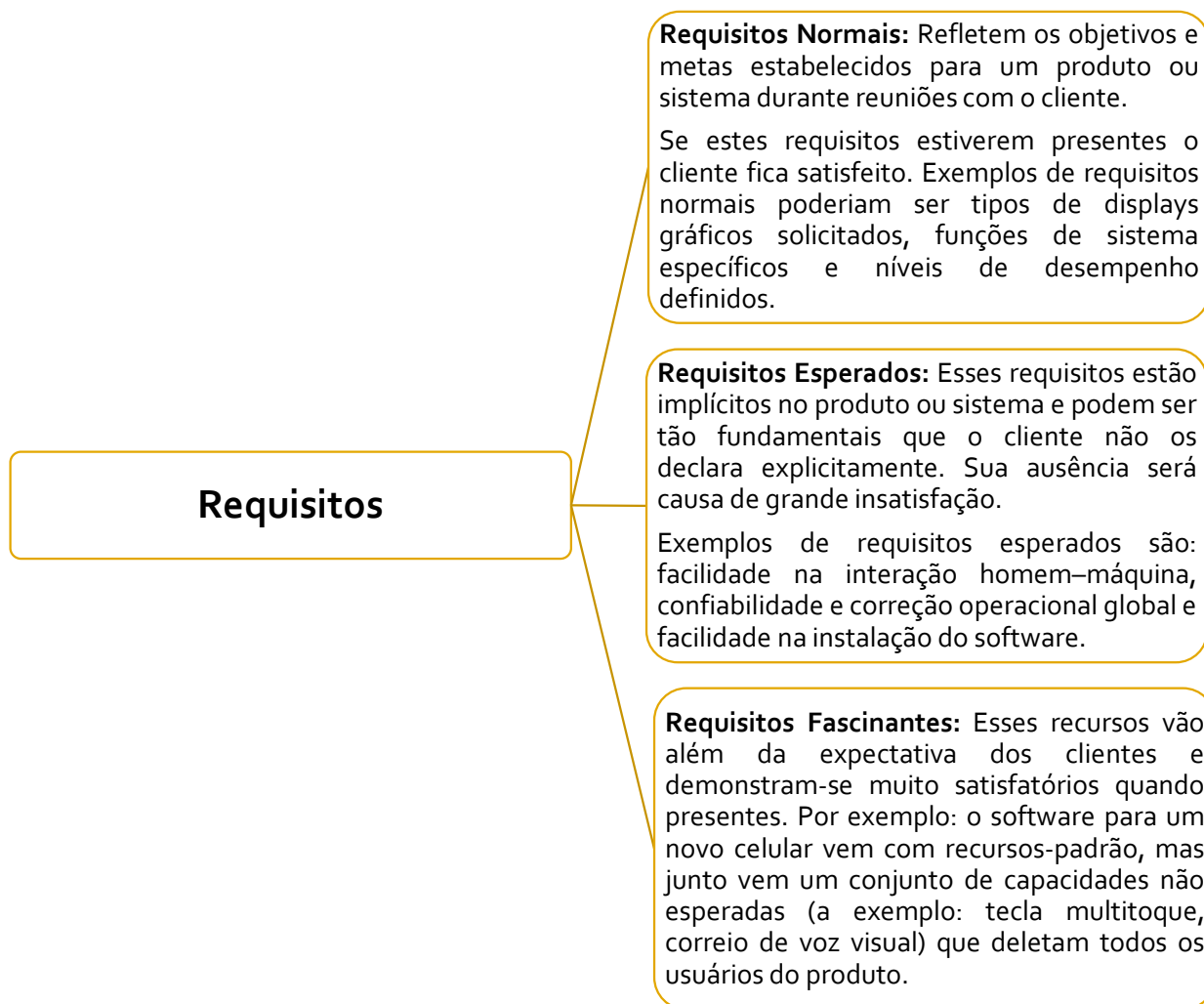
(O grupo gasta os 20 minutos seguintes refinando e expandindo os detalhes da função segurança domiciliar.)

Validação de Requisitos

Neste processo, valida-se a consistência, precisão, contextualização dos requisitos levantados no processo de identificação e descoberta de análise da regra de negócio do cliente. Envolvendo a revisão de todos os tipos de requisitos utilizados, validando assim o modelo.

Este passo é de extrema importância, visto que ele bem definido permite a não tendência de um projeto desastroso, cheio de falhas e com recursos não solicitado pelo cliente.

Ainda sobre os requisitos temos:



Especificação de Requisitos

As especificações, tem como objetivo obter um produto de software de excelente qualidade, que satisfaçam a real necessidade do cliente dentro do cronograma e orçamento aceito.

Podemos entender o requisito como uma função, restrição ou propriedade que deve ser fornecida, encontrada para atender e satisfazer à necessidade do usuário final do sistema. É comprovada que a maior parte dos problemas no desenvolvimento de sistemas, mesmo os de maior impacto negativos e mais onerosos, são frutos de uma Análise de Requisito errônea.

Cenários de Uso

À medida que os requisitos são levantados, uma visão geral das funções e características começa a ser materializada. Entretanto, é difícil progredir para as atividades de engenharia de software técnicas até que

entendamos como tais funções e características serão usadas por diferentes classes de usuários. Para tanto, os desenvolvedores e usuários podem criar um conjunto de cenários que identifiquem um roteiro de uso para o sistema a ser construído. Os cenários, normalmente chamados “casos de uso” fornecem uma descrição de como o sistema será utilizado.



Desenvolvimento de um cenário de uso preliminar

Cena: Sala de reuniões, na qual prossegue a primeira reunião de levantamento de requisitos.

Atores: Jamie Lazar, membro da equipe de software; Vinod Raman, membro da equipe de software; Ed Robbins, membro da equipe de software; Doug Miller, gerente da engenharia de software; três membros do Depto. de Marketing; um representante da Engenharia de Produto e um facilitador.

Conversa:

Facilitador: Andamos conversando sobre segurança de acesso à funcionalidade CasaSegura que poderá ser acessada via Internet. Gostaria de tentar algo. Vamos desenvolver um cenário de uso para acesso à função segurança domiciliar.

Jamie: Como?

Facilitador: Podemos realizar isso de várias maneiras, mas, por enquanto, gostaria de manter as coisas realmente informais. Conte-nos (ele aponta para um representante do Depto. de Marketing) como você imagina o acesso ao sistema.

Representante do Depto. de Marketing: Huum... Bem, esse é o tipo de coisa que eu faria se estivesse fora e tivesse que deixar alguém em casa, digamos uma empregada ou um encanador, que não teriam o código de acesso.

Facilitador (sorrindo): Isso que você falou é a razão para você fazê-lo... Diga-me como realmente faria isso.

Representante do Depto. de Marketing: Huum... A primeira coisa de que precisaria seria um PC. Entraria em um site que deveríamos manter para todos os usuários do CasaSegura. Forneceria meu nome de usuário e...

Vinod (interrompendo): A página teria de ser segura e criptografada para garantir que estamos seguros e...

Facilitador (interrompendo): Essas são informações adequadas, Vinod, porém são técnicas. Concentremo-nos apenas em como o usuário final usará essa capacidade. OK?

Vinod: Sem problemas.

Representante do Depto. de Marketing: Bem, como estava dizendo, entraria em um site e forneceria meu nome de usuário e dois níveis de senhas.

Jamie: O que acontece se eu esquecer minha senha?

Facilitador (interrompendo): Excelente observação, Jamie, mas não tratemos disso agora. Faremos um registro de sua observação e a chamaremos de exceção. Tenho certeza de que existirão outras.

Representante do Depto. de Marketing: Após eu introduzir as senhas, uma tela representando todas as funções do CasaSegura aparecerá. Eu selecionaria a função de segurança domiciliar. O sistema poderia solicitar que eu verificasse quem sou eu, digamos, perguntando meu endereço ou telefone ou algo do gênero. Em seguida, ele exibiria uma imagem do painel de controle do sistema de segurança com uma lista das funções que eu poderia executar — armar o sistema, desarmá-lo, desarmar um ou mais sensores. Suponho que ele também me permitiria reconfigurar zonas de segurança e outros itens similares, mas, não tenho certeza disso.

(Enquanto o representante do Depto. de Marketing continua falando, Doug faz anotações de maneira ininterrupta; essas formam a base para o primeiro cenário de uso informal. Como alternativa, poderia ser solicitado ao representante do Depto. de Marketing que escrevesse o cenário, mas isso seria feito fora da reunião.)

Artefatos do Levantamento de Requisitos

Os artefatos produzidos como consequência do levantamento de requisitos sofrerão variações de acordo com a dimensão do projeto:

Para a maioria dos sistemas, entre os artefatos, temos:

- Declarações das necessidades e das viabilidades.
- Declarações restritas ao escopo para o sistema ou produto.

- Lista de clientes, usuários e outros atores que participam do processo de levantamento de requisitos.
- Descrição do ambiente técnico do sistema.
- Lista de requisitos, preferencialmente organizada por funções, e as restrições de domínio que se aplicam a cada uma delas.
- Conjunto de cenários de uso que esclarecem o uso do sistema ou produto em diferentes condições operacionais.
- Melhores protótipos desenvolvidos para a melhor definição dos requisitos levantados.

Requisitos Funcionais

Na engenharia de software, um **requisito funcional** define uma função de um sistema de software ou seu componente. O requisito funcional representa o que o software faz, em termos de tarefas e serviços. Uma função é descrita como um conjunto de entradas, seu comportamento e as saídas. Os requisitos funcionais podem ser cálculos, detalhes técnicos, manipulação de dados e de processamento e outras funcionalidades específicas que definem o que um sistema, idealmente, será capaz de realizar.

Requisitos comportamentais descrevem todos os casos em que o sistema utiliza os requisitos funcionais, são extraídos dos casos de uso. Além disso, os requisitos funcionais são suportados por requisitos não funcionais conhecidos como "**requisitos de qualidade**", que impõem restrições sobre o projeto ou execução tais como requisitos de desempenho, segurança ou confiabilidade. O plano para a implementação dos requisitos funcionais é detalhado no projeto do sistema. Entretanto, o plano para a implementação de requisitos não funcionais é detalhado na arquitetura do sistema.

Definidos na engenharia de requisitos, os requisitos funcionais especificam resultados particulares de um sistema. Isto deve ser contrastado com requisitos não funcionais, os quais especificam características gerais, tais como custo e confiabilidade. Os requisitos funcionais fazem parte da arquitetura do aplicativo de um sistema, enquanto os requisitos não funcionais denotam a arquitetura técnica de um sistema.

Qualidades dos requisitos funcionais:

- **Prioridade:** é de extrema importância o conhecimento da prioridade do requisito, devido a esta característica ser usada no plano de iterações, em cada iteração, os requisitos abertos com maior prioridade são detalhados e desenvolvidos. Neste sentido, o desenvolvimento ágil garante que o cliente receba o que ele mais precisa em um dado momento.
- **INVESTE – Independent, Negotiable, Valuable, Estimable, Small and Testable:** devem ser Independentes, Negociáveis, Valoráveis, Estimáveis, Pequenos e Testáveis.
- **Understandability:** Estes requisitos devem ser escritos na língua que o cliente entenda.

- **Forward Traceable:** permite que o requisito possua rastreabilidade entre o código fonte e casos de testes. Caso seja alterado um caso de testes identificaremos em qual código será refletido e vice-versa. Isto de certo modo nos lembra a dependência, muito usada na programação orientada a objeto.

Requisitos não Funcionais

Este modelo de requisitos é relacionado ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenção e tecnologias envolvidas. Eles identificam as funcionalidades que serão entregues ao usuário do software.

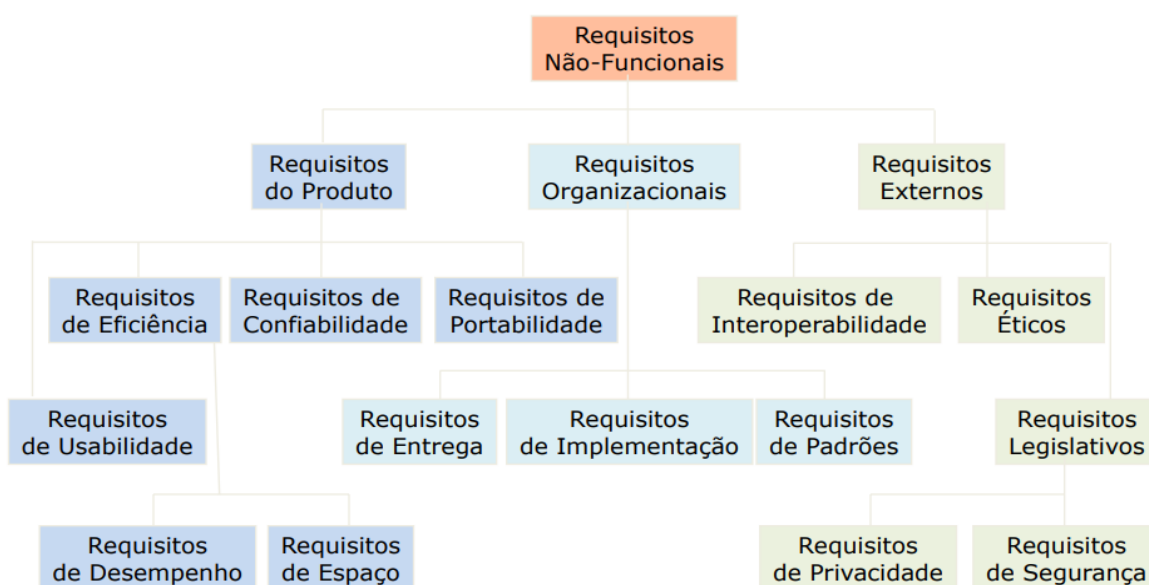
Características:

- Demonstram qualidade acerca dos serviços ou funções disponibilizadas pelo sistema. Ex.: tempo, o processo de desenvolvimento, padrões, etc.
- Surgem conforme a necessidade dos usuários em razão de orçamento e outros fatores.
- Podem estar relacionados à confiabilidade, tempo de resposta e espaço nas mídias de armazenamento disponíveis.
- Caso ocorra falha do não atendimento a um requisito não funcional, isto poderá tornar todo o sistema ineficaz. Ex.: requisito confiabilidade em um sistema de controle aéreo.

Classificação:

- **Requisitos de produtos:** Requisitos que especificam o comportamento do produto. Ex. portabilidade; tempo na execução; confiabilidade, mobilidade, etc.
- **Requisitos de usabilidade ("facilidade de uso"):** Usuários deverão operar o sistema após um determinado tempo de treinamento.
- **Requisito de Eficiência:** O sistema deverá processar inúmeras requisições por minuto.
- **Requisito de confiabilidade:** O sistema deverá ter alta disponibilidade, por exemplo em 99% do tempo.
- **Requerimento de portabilidade:** O aplicativo deverá ser executado em qualquer plataforma.
- **Requisitos organizacionais:** Estes são decorrentes de políticas e procedimentos corporativos, por exemplo, padrões de infraestrutura, políticas organizacionais da empresa.
- **Requisitos de entrega:** Um breve e bom exemplo que explica esse conceito é o de alguns relatórios que devem ser entregues rotineiramente em determinadas datas.
- **Requisito de implementação:** Por exemplo, o sistema deverá ser desenvolvido em linguagem de programação Python.
- **Requisito de padrões:** O uso de programação orientada a objeto, este é um exemplo clássico.

- **Requisitos externos:** Este é decorrente de fatores externos ao sistema, também ao processo de desenvolvimento.
- **Requisito de interoperabilidade:** Um exemplo prático é o do sistema que deverá comunicar-se com o banco de dados Oracle.
- **Requisitos éticos:** O sistema não apresenta aos usuários dados de características privativa.
- **Requisitos legais:** O sistema desenvolvido deverá atender as normas legais, como padrões, leis, normas, regras etc.



12

Requisitos de Usuários de Sistema

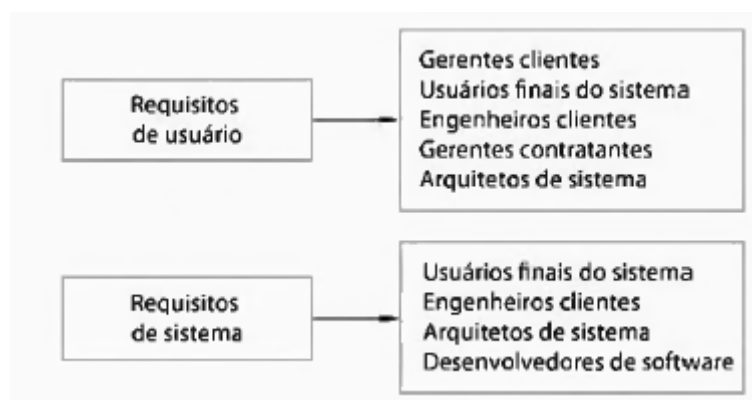
Definição de requisitos de usuário

1. O MHC-PM5 deve gerar relatórios gerenciais mensais que mostrem o custo dos medicamentos prescritos por cada clínica durante aquele mês.

Especificação de requisitos de sistema

- 1.1 No último dia útil de cada mês deve ser gerado um resumo dos medicamentos prescritos, seus custos e as prescrições de cada clínica.
- 1.2 Após 17:30h do último dia útil do mês, o sistema deve gerar automaticamente o relatório para impressão.
- 1.3 Um relatório será criado para cada clínica, listando os nomes dos medicamentos, o número total de prescrições, o número de doses prescritas e o custo total dos medicamentos prescritos.
- 1.4 Se os medicamentos estão disponíveis em diferentes unidades de dosagem (por exemplo, 10 mg, 20 mg), devem ser criados relatórios separados para cada unidade.
- 1.5 O acesso aos relatórios de custos deve ser restrito a usuários autorizados por uma lista de controle de gerenciamento de acesso.

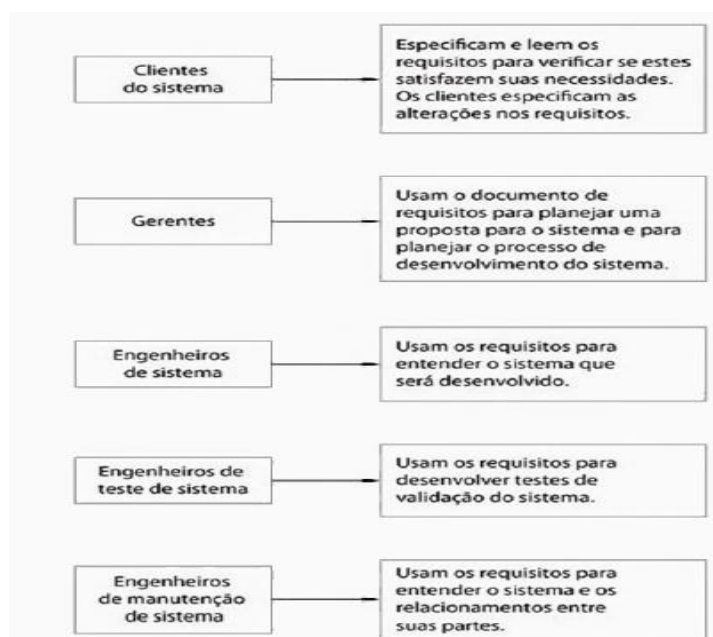
Tipos de Especificação de Requisitos



Métricas para Especificar Requisitos Não Funcionais

Propriedade	Medida
Velocidade	Transações processadas/segundo Tempo de resposta de usuário/evento Tempo de atualização de tela
Tamanho	Megabytes Número de chips de memória ROM
Facilidade de uso	Tempo de treinamento Número de frames de ajuda
Confiabilidade	Tempo médio para falha Probabilidade de indisponibilidade Taxa de ocorrência de falhas Disponibilidade
Robustez	Tempo de reinício após falha Percentual de eventos que causam falhas Probabilidade de corrupção de dados em caso de falha
Portabilidade	Percentual de declarações dependentes do sistema-alvo Número de sistemas-alvo

Usuários do Processo



Estrutura de Documento de Requisitos

Capítulo	Descrição
Prefácio	Deve definir os possíveis leitores do documento e descrever seu histórico de versões, incluindo uma justificativa para a criação de uma nova versão e um resumo das mudanças feitas em cada versão.
Introdução	Deve descrever a necessidade para o sistema. Deve descrever brevemente as funções do sistema e explicar como ele vai funcionar com outros sistemas. Também deve descrever como o sistema atende aos objetivos globais de negócio ou estratégicos da organização que encomendou o software.
Glossário	Deve definir os termos técnicos usados no documento. Você não deve fazer suposições sobre a experiência ou o conhecimento do leitor.
Definição de requisitos de usuário	Deve descrever os serviços fornecidos ao usuário. Os requisitos não funcionais de sistema também devem ser descritos nessa seção. Essa descrição pode usar a linguagem natural, diagramas ou outras notações compreensíveis para os clientes. Normas de produto e processos que devem ser seguidos devem ser especificados.
Arquitetura do sistema	Deve apresentar uma visão geral em alto nível da arquitetura do sistema previsto, mostrando a distribuição de funções entre os módulos do sistema. Componentes de arquitetura que são reusados devem ser destacados.
Especificação de requisitos do sistema	Deve descrever em detalhes os requisitos funcionais e não funcionais. Se necessário, também podem ser adicionados mais detalhes aos requisitos não funcionais. Interfaces com outros sistemas podem ser definidas.
Modelos do sistema	Pode incluir modelos gráficos do sistema que mostram os relacionamentos entre os componentes do sistema, o sistema e seu ambiente. Exemplos de possíveis modelos são modelos de objetos, modelos de fluxo de dados ou modelos semânticos de dados.
Evolução do sistema	Deve descrever os pressupostos fundamentais em que o sistema se baseia, bem como quaisquer mudanças previstas, em decorrência da evolução de hardware, de mudanças nas necessidades do usuário etc. Essa seção é útil para projetistas de sistema, pois pode ajudá-los a evitar decisões capazes de restringir possíveis mudanças futuras no sistema.
Apêndices	Deve fornecer informações detalhadas e específicas relacionadas à aplicação em desenvolvimento, além de descrições de hardware e banco de dados, por exemplo. Os requisitos de hardware definem as configurações mínimas ideais para o sistema. Requisitos de banco de dados definem a organização lógica dos dados usados pelo sistema e os relacionamentos entre esses dados.
Índice	Vários índices podem ser incluídos no documento. Pode haver, além de um índice alfabético normal, um índice de diagramas, de funções, entre outros pertinentes.

Questões comentadas pelo professor

1. (CESPE - 2011 - MEC - Gerente de Projetos)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A engenharia de *software*, disciplina relacionada aos aspectos da produção de *software*, abrange somente os processos técnicos do desenvolvimento de *software*.

() Certo.

() Errado.

RESOLUÇÃO:

Muito pelo contrário, a Engenharia de Software abrange todo o processo de produção de um projeto de software.

Resposta: Errado

2. (CESPE - 2013 - TRT - 10ª REGIÃO (DF e TO) - Analista Judiciário - Tecnologia da Informação)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A engenharia de *software* engloba processos, métodos e ferramentas. Um de seus focos é a produção de *software* de alta qualidade a custos adequados.

() Certo.

() Errado.

RESOLUÇÃO:

Assertiva correta, vimos lá no início do nosso material que existe uma disciplina responsável pela qualidade do serviço entregue, esta é a Qualidade de Software.

Resposta: Certo

3. (CESPE - 2010 - TRE-BA - Analista Judiciário - Análise de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Entre os desafios enfrentados pela engenharia de *software* estão lidar com sistemas legados, atender à crescente diversidade e atender às exigências quanto a prazos de entrega reduzidos.

- () Certo.
() Errado.

RESOLUÇÃO:

Pessoal, boa parte dos sistemas existentes no mercado, são sistemas legados: sistemas antigos que ainda estão em produção, entretanto, na maioria das vezes atendem à demanda com diversas falhas, foram baseados em tecnologias antigas, a manutenção é inviável e em sua maioria muito cara, sem falar que nem sempre possuem manutenção e correções de bugs pelos fabricantes, o que torna complexo o desenvolver das atividades através destes softwares.

Resposta: Certo

4. (CESPE - 2010 - DETRAN-ES - Analista de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Segundo princípio da engenharia de *software*, os vários artefatos produzidos ao longo do seu ciclo de vida apresentam, de forma geral, nível de abstração cada vez menor

- () Certo.
() Errado.

RESOLUÇÃO:

Amigos, a assertiva é dada como correta, mas pessoalmente acho isto meio que relativo, sabe porquê? Dependendo do modelo de desenvolvimento que usamos ou ciclo de vida que é utilizado, existem casos em que o cliente não consegue passar de forma clara alguns processos de requisito, o que lá na frente pode gerar um grande problema, ou seja, isto é muito relativo.

Resposta: Certo

5. (CESPE - 2010 - TRE-BA - Técnico Judiciário - Programação de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Em uma visão restritiva, muitas pessoas costumam associar o termo *software* aos programas de computador. *Software* não é apenas o programa, mas também todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente.

A respeito de engenharia de *software*, julgue os itens

A engenharia de *software* está relacionada com todos os aspectos da produção de *software*, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação. A engenharia de

sistemas diz respeito aos aspectos do desenvolvimento e da evolução de sistemas complexos, nos quais o *software* desempenha um papel importante.

() Certo.

() Errado.

RESOLUÇÃO:

Correto, a Engenharia de Software é a área que percorre e deve se fazer presente por todo o processo do desenvolvimento do sistema.

Resposta: Certo

6. (CESPE - 2013 - TRT - 10ª REGIÃO (DF e TO) - Analista Judiciário - Tecnologia da Informação)

Com relação a conceitos gerais da engenharia de *software*, julgue.

As atividades fundamentais relacionadas ao processo de construção de um *software* incluem a especificação, o desenvolvimento, a validação e a evolução do *software*.

() Certo.

() Errado.

RESOLUÇÃO:

Estas são as atividades básicas para que a construção de um software seja bem sucedida, isto é, se todos os passos forem executados de maneira correta.

Resposta: Certo

7. (CESPE - 2013 - MPOG - Técnico de Nível Superior - V - Categoria Profissional 7)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A evolução de um *software* é realizada ao ser testados, integrados e aceitos os componentes individuais.

() Certo.

() Errado.

RESOLUÇÃO:

Na verdade, a evolução de software, ocorre quando se alteram os atuais sistemas de software para atender aos novos requisitos.

Resposta: Errado

8. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Engenharia de software não está relacionada somente aos processos técnicos de desenvolvimento de softwares, mas também a atividades como gerenciamento de projeto e desenvolvimento de ferramentas, métodos e teorias que apoiem a produção de softwares.

() Certo.

() Errado.

RESOLUÇÃO:

No processo da engenharia de software, não se preocupa apenas com os processos técnicos do desenvolvimento. Também inclui atividades como gerenciamento de projeto de software e desenvolvimento de ferramentas, métodos e teorias para apoiar a produção do produto.

Resposta: Certo

9. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Sistemas que incluem softwares são classificados em duas categorias: sistemas técnicos embasados em computadores e sistemas sócio técnicos. Os primeiros incluem componentes de hardware, software, pessoas, procedimentos e processos; os segundos são regidos pelas políticas e regras organizacionais.

() Certo.

() Errado.

RESOLUÇÃO:

Sommerville, aborda em seu livro: Um sistema é o conjunto intencional de componentes inter-relacionados que funcionam juntos para atingir certo objetivo. São duas as categorias de sistemas que incluem softwares: Sistemas técnicos baseados em computadores e Sistemas sociotécnicos.

Resposta: Errado

10. (CESPE - 2013 - STF - Analista Judiciário - Análise de Sistemas de Informação)

Com referência a aspectos diversos de engenharia de software, julgue os itens subsecutivos.

Engenharia de *software* é um ramo da ciência da computação que se preocupa com o desenvolvimento de códigos que sejam confiáveis, seguros, baratos e que, quando executados, funcionem e tenham o desempenho desejado.

() Certo.

() Errado.

RESOLUÇÃO:

O sempre barato, nos entregou a questão de graça, pois essa história de projeto de software barato, sabemos que isso não existe.

Resposta: Errado

11. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Com referência a aspectos diversos de engenharia de software, julgue os itens subsecutivos.

O ciclo de vida clássico da engenharia de software tem sofrido críticas dos especialistas. Uma delas é o fato de o cliente ter de declarar todos os requisitos explicitamente na primeira parte do projeto, gerando insegurança. Uma técnica utilizada para minimizar esse problema é conhecida como:

- A) Análise por ponto de função.
- B) Casos de teste.
- C) Linha básica ou baseline.
- D) Projeto de software.
- E) Prototipação.

RESOLUÇÃO:

A Prototipação ou Prototipagem é, a construção de um exemplar do que foi entendido dos requisitos capturados do cliente. Ou seja, praticamente iremos desenvolver algo para que o cliente possa usar como se fosse inicialmente o sistema. Pode ser considerado um ciclo de vida ou pode ser usado como ferramenta em outros ciclos de vida. Esta modalidade trouxe segurança ao processo da Engenharia de Software usando o ciclo da prototipação.

Resposta: E

12. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

A análise por pontos de função utiliza diversas características para estimar o tamanho de um software. Das características abaixo, indique a que **NÃO** afeta a contagem nesse tipo de métrica.

- A) Desempenho.
- B) Necessidade de backup.
- C) Necessidade de testes.
- D) Necessidade de comunicação de dados.
- E) Número de entradas do usuário.

RESOLUÇÃO:

A necessidade de testes, não oferece influência em nenhum momento em que são realizadas, a exemplo: no período de análise de pontos de função.

Resposta: C

13. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Devido ao aumento da demanda por aplicações que funcionem 24 horas por dia na Internet, um software deve ser capaz de manter-se em operação após uma determinada falha. A estratégia de teste que melhor garante essa característica é o(a):

- A) Teste de estresse.
- B) Teste de recuperação.
- C) Teste unitário.
- D) Debugging.
- E) Simulação combinatória.

RESOLUÇÃO:

Alunos, o teste ideal é o Teste de Recuperação, pois por meio dele conseguimos verificar a densidade e robustez e o próprio tempo de capacidade de um determinado software de retornar em caso de uma queda ou falha.

Resposta: C

14. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Na análise estruturada, as duas ferramentas de modelagem que identificam de modo mais adequado a estrutura dos dados são:

- A) Diagrama de entidades-relacionamentos e dicionário de dados.
- B) Diagrama de transições de estado e diagrama de fluxo de dados.
- C) Diagrama de fluxo de dados e dicionário de dados.
- D) Diagrama de fluxo de dados e diagrama de entidades relacionamentos.
- E) Dicionário de dados e diagrama de transições de estado.

RESOLUÇÃO:

O Diagrama de Fluxo de Dados são os responsáveis pela parte do fluxo de dados, também do processo dos dados. Os Diagramas de Transições de Estado, focam nos eventos e nos estados do sistema. O Dicionário de Dados, é responsável pela descrição de cada campo de uma tabela no banco de dados. DER, conhecido também como Diagrama de Entidade Relacionamento, descreve os dados e as suas colaborações entre eles no sistema.

Resposta: A

15. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Um processo administrativo numa repartição pública poderá estar nos seguintes estados: aberto, na carga de um determinado funcionário, em trâmite ou arquivado. A ferramenta de modelagem que representa de modo mais adequado as regras de passagem de um estado para outro é:

- A) Diagrama de fluxo de dados.
- B) Diagrama de transições de estado.
- C) Diagrama entidades-relacionamentos.
- D) Dicionário de dados.
- E) Especificação de processos.

RESOLUÇÃO:

Isto mesmo, a ferramenta de modelagem que representa a passagem de um modelo de um estado para o outro é o Diagrama de Transição, e este faz parte do UML.

Resposta: B

16. (CESPE - 2010 - SERPRO - Analista Desenvolvimento de Sistemas | CESPE - 2010 - SERPRO - Analista - Desenvolvimento de Sistemas)

A engenharia de software é uma disciplina do conhecimento humano que aplica princípios da engenharia ao desenvolvimento de software. No entanto, existem diferenças significativas entre as engenharias clássicas — mecânica, civil, elétrica — e a engenharia de software, muitas delas decorrentes das diferenças de natureza entre o produto ou sistema resultante da atividade das engenharias clássicas e a engenharia de software. A engenharia de software desdobrou-se em várias áreas especializadas, como as áreas de requisitos de software, de análise e projeto de software, de implementação de software, de testes de software, de gestão de configuração, entre outras. Na interface entre as engenharias clássicas e a de software insere-se a engenharia de sistemas, abordada colateralmente na ISO/IEC 12207.

A homologação de um *software* é uma atividade pertinente aos processos de aquisição e não aos processos de garantia de qualidade, sendo ainda diretamente relacionada ao ciclo de vida de sistemas e não apenas de *software*.

☐ Certo.

☐ Errado.

RESOLUÇÃO:

A homologação do sistema, realmente faz parte do processo de aquisição.

Resposta: Certo

17. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação à engenharia de software, julgue os itens seguintes.

A engenharia de software define quatro fases para o ciclo de vida de um sistema: iniciação, elaboração, construção e transição, sendo essa última responsável pela homologação dos artefatos junto ao cliente.

☐ Certo.

☐ Errado.

RESOLUÇÃO:

As fases do ciclo de vida de um sistema são as seguintes: Definição, Desenvolvimento, Operação, Retirada.

Resposta: Errado

18. (CESPE - 2010 - TRE-BA - Técnico Judiciário - Programação de Sistemas)

Em uma visão restritiva, muitas pessoas costumam associar o termo *software* aos programas de computador. *Software* não é apenas o programa, mas também todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente. A respeito de engenharia de *software*, julgue os itens.

A engenharia de *software* está relacionada com todos os aspectos da produção de *software*, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação. A engenharia de sistemas diz respeito aos aspectos do desenvolvimento e da evolução de sistemas complexos, nos quais o *software* desempenha um papel importante.

() Certo.

() Errado.

RESOLUÇÃO:

Em meu humilde conceito é a melhor definição da Engenharia de Software e seus conceitos, abrangendo todo o processo e até mesmo depois de finalizado e entregue o produto ao cliente final com sua integridade e satisfação.

Resposta: Certo

19. (CESPE - 2008 - INPE - Tecnologista Júnior - TS12)

Com relação aos conceitos de engenharia de software, julgue os próximos itens.

A técnica de inspeção de software é mais custosa do que os testes.

() Certo.

() Errado.

RESOLUÇÃO:

Amigos, as técnicas de inspeções são muitas mais rápidas e baratas do que o teste de software, a exemplo uma técnica de inspeção do software pode ser no momento de depuração ou depuração do código fonte do sistema.

Resposta: Errado

20. (CESPE - 2008 - INPE - Tecnologista Júnior - TS12)

Com relação aos conceitos de engenharia de software, julgue os próximos itens.

As abstrações de dados estão entre os tipos de módulos que podem ser criados durante o processo de modularização de um programa.

() Certo.

() Errado.

RESOLUÇÃO:

Para que seja criado cada módulo de um projeto de sistemas por meio da engenharia de software, há a necessidade básica de criarmos modelos de abstrações para realizarmos as abstrações e obtermos as informações para o desenvolver do projeto.

Resposta: Certo

21. (CESPE - 2014 - TC-DF - Analista de Administração Pública - Sistemas de TI)

Julgue os itens a seguir, com relação à engenharia de software.

O modelo de negócio é definido como um desenho das atividades sequenciadas geradas por entradas e que geram também saídas, apoiadas por artefatos específicos

() Certo.

() Errado.

RESOLUÇÃO:

A assertiva está errada, pois a definição do modelo de negócio é a descrição da lógica de criação, entregue e captura de valor por parte de uma organização.

Resposta: Errado

22. (CESPE - 2014 - ANATEL - Analista Administrativo - Desenvolvimento de Sistemas)

A respeito de engenharia de *software*, julgue os próximos itens.

Em sistemas em tempo real orientados a eventos, é adequada a utilização de um modelo de controle orientado à interrupção.

() Certo.

() Errado.

RESOLUÇÃO:

Os modelos orientados a interrupções, são descritos para sistema de tempo real.

Resposta: Certo

23. (CESPE - 2014 - ANATEL - Analista Administrativo - Desenvolvimento de Sistemas)

A respeito de engenharia de *software*, julgue os próximos itens.

A recursividade é uma técnica que pode ser utilizada na implementação de sistemas de lógica complexa, com a finalidade de minimizar riscos de ocorrência de defeitos no *software*.

() Certo.

() Errado.

RESOLUÇÃO:

O que pode minimizar os erros, também chamados defeitos de softwares, são os testes de softwares.

Resposta: Errado

24. (CESPE - 2013 - BACEN - Analista - Análise e Desenvolvimento de Sistemas)

A engenharia de *software* engloba, além dos processos técnicos de desenvolvimento de *software*, atividades de gerenciamento de projeto de *software* e desenvolvimento de ferramentas, métodos e procedimento para apoiar a produção de *software*. Oferece uma base para a construção de *software* de alta qualidade e produtividade. Com relação à engenharia de *software*, julgue o item que se segue.

Procedimento é o conjunto de tarefas que incluem os planejamentos e estimativas de projeto, análise de requisitos de sistema, modelagem de dados, arquitetura, codificação, teste e manutenção.

() Certo.

() Errado.

RESOLUÇÃO:

Na verdade, o procedimento é como será executada algumas tarefas em determinados processos.

Resposta: Errado

25. (CESPE - 2013 - BACEN - Analista - Análise e Desenvolvimento de Sistemas)

Julgue o item a seguir, relativo a análise e projeto de *software*.

A linguagem ubíqua utiliza termos que fazem parte das discussões entre os especialistas de negócio e as equipes de desenvolvimento, os quais devem utilizar a mesma terminologia na linguagem falada e no código.

() Certo.

() Errado.

RESOLUÇÃO:

A linguagem é como uma linguagem universal usada em meio a Engenharia de Software, assim, conseguimos estreitar o abismo técnico entre programadores e outras categorias de especialistas envolvido no negócio, facilitando a comunicação e estreitando os problemas na troca de informações.

Resposta: Certo

26. (CESPE - 2013 - SEGESP-AL - Perito Criminal - Tecnologia da Informação)

Em relação aos conceitos de engenharia de *software* e ciclo de vida de um *software*, julgue o item a seguir.

O desenvolvimento de produto de *software* por meio da utilização da técnica de prototipação consiste na execução das atividades de desenvolvimento de *software* em uma sequência ordenada, sendo as principais atividades do modelo requisitos de sistema; requisitos de *software*; análise; projeto de programa; codificação; teste; e operação.

() Certo.

() Errado.

RESOLUÇÃO:

Muito pelo contrário, a prototipação é o conceito em que criamos um software com características do que será desenvolvido no projeto, depois entregue ao cliente para que possa ser feita avaliações e modificações, muitas das vezes o protótipo torna-se o próprio software que outrora fora desenvolvido, já em outros casos, ele é descartado.

Resposta: Errado

27.(FCC - 2015 - TRE-AP - Técnico Judiciário - Programação de Sistemas)

Um Técnico participou do levantamento de requisitos de um novo sistema do Tribunal. Devidamente autorizado, ele se inseriu no ambiente de trabalho em que o sistema seria utilizado e observou o trabalho diário, anotando as tarefas reais. Seu principal objetivo era descobrir requisitos de sistema implícitos, que refletissem os processos reais nos quais as pessoas estão envolvidas, ao invés de processos formais. Além destes requisitos, ele também coletou os requisitos derivados da cooperação e conscientização das atividades de outras pessoas envolvidas. O Técnico estava colocando em prática a técnica de levantamento de requisitos denominada.

- A) VORD (*Viewpoint-Oriented Requirements Definition*) ou Definição de Requisitos Orientada a Ponto de Vista.
- B) Etnografia.
- C) JAD (*Joint Application Design*).
- D) *Workshop* ou Oficinas de Trabalho.
- E) Prototipagem.

RESOLUÇÃO:

Amigos, vamos aprender um pouco mais com essa questão, entre as técnicas de levantamentos de requisitos as principais são: 1) **Brainstorming**: sem julgamentos ou análises, ambiente descontraído e informal, para novos produtos. 2) **JAD**: cooperação, entendimento e trabalho em grupo, 3) **Análise de documentos quantitativos**: formulários e relatórios, 4) **Reunião**: licitação de requisitos em grupo, 5) **Prototipagem**: para atrair aspectos críticos quando não houver domínio mínimo da aplicação. 6) **Entrevista**: conversa para extrair tópicos importantes. 7) **Questionários**: questões subjetivas e objetivas. 8) **Observação**: comportamento e o ambiente. 9) **Viewpoint-Oriented Requirements Definition**: pontos de vista dos usuários, analisando as diferenças e similaridades. 10) **Etnografia**: para entender a organização, sua cultura e o objetivo pelo inserir de analista no ambiente de trabalho. 11) **Caso de Uso**: comportamento externo de um sistema descrevendo ações para produzir um resultado observável por um ator, através de interação entre um ator (usuário, outro sistema computacional ou um dispositivo) e um sistema.

Resposta: B

28.(CESGRANRIO - 2012 - LIQUIGÁS - Profissional Júnior - Administração de Banco de Dados)

Ao levantar os requisitos necessários para a implantação de um workflow em uma empresa, a equipe responsável percebeu que o requisito mais importante era a facilidade de definir o processo, sendo a flexibilidade mais relevante que a produtividade. Também era desejado que as definições de processos fossem criadas com formulários.

Nesse caso, o tipo de workflow mais adequado é o:

- A) Ad hoc.
- B) Administrativo.
- C) Autônomo.
- D) Colaborativo.
- E) Produção.

RESOLUÇÃO:

Pessoal, colocarei aqui na resolução os tipos de Workflow para que você não caia mais em pegadinhas deste tipo:

➤ **Ad hoc Workflow:** *neste tipo, podem haver mudanças nas regras, mesmo que a operação do processo tenha iniciado. Este modelo de trabalho é indicado para grupos formados por membros que executam tarefas de forma individual.*

➤ **Workflow de Produção:** *neste modelo, o Workflow é indicado para grupos envolvidos com um grande volume de dados, recursos financeiros e políticas de negócios em grande escala. Em grupos com essas características os procedimentos já estão bem definidos e não há possibilidades para mudanças, alterações ou criação de novas regras.*

➤ **Workflow Administrativo:** boa parte dos profissionais, consideram este tipo como um meio-termo entre o Ad hoc e o Workflow de Produção. É ideal para rotinas administrativas e demais grupos envolvidos em atividades constantes, previsíveis, pautadas por regras simples de coordenação. Tarefas que envolvam formulários e tratamento de documentos também se adequam bem a este Workflow.

Resposta: B

29.(CESGRANRIO - 2006 - Petrobrás - Analista de Sistemas Pleno - Engenharia de Software | CESGRANRIO - 2006 - Petrobrás - Analista de Sistemas Pleno - Processos)

Sobre a Análise e o Gerenciamento de Requisitos, é **FALSO** afirmar que:

- A) Quanto mais tarde for identificado um problema na análise de requisitos, maior será o custo com o retrabalho.

- B) A elicitação é o processo de identificação e entendimento das necessidades e restrições dos usuários, enquanto que a especificação é o processo de formalização das necessidades e restrições dos usuários em requisitos funcionais de software.
- C) Na análise de requisitos o cliente utiliza as melhores práticas de engenharia de requisitos na tarefa de descrever suas necessidades.
- D) O gerenciamento de requisitos corresponde ao conjunto de atividades que auxilia a equipe do projeto a identificar, controlar e rastrear os requisitos, bem como a fazer as alterações nos requisitos durante o projeto.
- E) O gerenciamento de requisitos implica a alteração, inclusão e/ou exclusão de requisitos ao produto de software, o que pode levar a alterações de prazos, de recursos humanos, de equipamentos e de tecnologia.

RESOLUÇÃO:

Assertiva errada, como pode o cliente usar das melhores práticas, se é ele quem precisa de ferramentas e técnicas oferecida pela análise para resolver o seu problema? Sem cabimento!

Resposta: C

30.(CESPE - 2004 - Polícia Federal - Perito Criminal Federal - Informática)

Considere que se deseja desenvolver um sistema para controle de caixa de supermercado tendo como base um computador que registra os produtos vendidos, interagindo com dispositivos de entrada e saída tais como impressora, teclado e leitora de código de barras. Esse sistema deve interagir também com o operador do caixa e com um banco de dados do estabelecimento. A partir dessas informações, julgue os itens que se seguem.

A descrição informal do que o sistema deve fazer, tal como ler código de barras, identificar o produto e calcular o total da compra, faz parte da especificação de requisitos do programa.

- () Certo.
- () Errado.

RESOLUÇÃO:

Assertiva correta, no processo de levantamento de requisitos são levantadas as regras de negócios da empresa, como todo o processo ou fluxo de trabalho, quanto mais completa a análise menor são as possibilidades de erros na execução do processo assistida pelo sistema.

Resposta: Certo

31.(CESGRANRIO - 2008 - BR Distribuidora - Analista de Sistemas Júnior - Processos de Negócios)

Analise as afirmativas a seguir, sobre requisitos em projetos de software.

I - O rastreamento de requisitos é de grande importância para conduzir análises de impacto quando há mudanças em requisitos.

II - O acrônimo FURPS+ se refere aos requisitos não funcionais das categorias de Feasibility, Usability, Reliability, Performance e Supportability.

III - Um requisito pode conter, além da especificação, atributos que sirvam ao seu gerenciamento.

IV - Casos de uso são descrições da interação entre um ator e o sistema e, portanto, especificam apenas requisitos funcionais.

Estão corretas **APENAS** as afirmativas

- A) I e II.
- B) I e III.
- C) II e III.
- D) II e IV.
- E) III e IV.

RESOLUÇÃO:

Pessoal, vamos aos erros da questão: *O acrônimo se refere ao requisito funcional Funcionalidade, aos requisitos não funcionais Usabilidade, Confiabilidade (Reliability), Desempenho (Performance) e Suportabilidade, e também a outros requisitos não funcionais, o + ao final do acrônimo.*

Resposta: B

32.(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.

RESOLUÇÃO:

Conforme abordado na aula, estas são técnicas de abordagem utilizadas na obtenção de requisitos.

Resposta: D

33.(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 afirmar APENAS em:

- A) I.
- B) II.
- C) III.
- D) I e III.
- E) II e III.

RESOLUÇÃO:

Requisitos do usuário, descrevem os requisitos funcionais, não funcionais e de domínio numa linguagem que o usuário final possa entender. Requisitos do sistema descrevem os requisitos de usuário de forma mais detalhada, usando-se termos técnicos e diagramas. Nestes diagramas podemos citar os Diagramas de Caso de Uso, que mostra em fluxo como o sistema em seu comportamento no momento da interação entre atores, assim como no fluxo de execução.

Resposta: E

34.(FCC - 2008 - TRT - 18ª Região (GO) - Analista Judiciário - Tecnologia da Informação)

Considerando que a especificação dos requisitos pode não ser completa durante o estágio de Análise de Requisitos, em razão da imaturidade de conhecimento de clientes e desenvolvedores, é recomendável que a análise e modelagem dos requisitos tenham uma abordagem.

- A) operacional, somente.
- B) bottom-up.
- C) conceitual, somente.
- D) iterativa.
- E) estratégica, somente.

Resposta: D

35.(FCC - 2010 - DPE-SP - Agente de Defensoria - Programador)

Na Engenharia de Requisitos, dentre passos a serem seguidos para elicitação de requisitos é INCORRETO: operacional, somente.

- A) Solicitar participação de várias pessoas para que os requisitos sejam definidos a partir de diversos pontos de vista.
- B) Identificar regras de domínio que limitam a funcionalidade ou desempenho do sistema ou produto que será construído.
- C) Definir um ou mais métodos de elicitação de requisitos.
- D) Selecionar as pessoas sem preconceitos organizacionais para auxiliar a especificar os requisitos.
- E) Identificar claramente a justificativa de existência para cada requisito registrado; identificar requisitos ambíguos que serão candidatos a prototipação.

RESOLUÇÃO:

Novamente uma questão polêmica e meia que subjetiva, quando fala em pegar as pessoas sem preconceitos organizacionais é meio perigoso, entretanto, este modelo funciona bem somente na teoria dos livros.

Resposta: D

36.(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.

RESOLUÇÃO:

O mecanismo de validação, nada mais é que, uma revisão técnica do que foi levantado, por isso, assertiva a ser escolhida é a letra E.

Resposta: E

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

No âmbito da Engenharia de *Software*, um cenário de constantes mudanças políticas como as que ocorrem em uma aplicação governamental, por exemplo, propõe a especificação de um elemento de *software* que é o:

- A) Requisito inconsistente.
- B) Requisito volátil.
- C) Processo indefinido
- D) Escopo reverso.
- E) Escopo inconsistente.

RESOLUÇÃO:

Para entender melhor a questão, você precisa conhecer o conceito do item escolhido como a opção certa.

Requisitos Voláteis – são requisitos que se modificam durante o desenvolvimento ou quando o sistema está em uso. Exemplo: Requisitos resultantes de políticas governamentais.

Resposta: B

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

No processo de elicitação e análise de requisitos, a técnica pela qual o analista, como recurso, insere-se no ambiente de trabalho onde o sistema será usado, observando e registrando a rotina diária dos envolvidos para descobrir requisitos implícitos do sistema que reflitam os processos reais e não os formais, denomina-se:

- A) Etnografia.
- B) Análise de Dados.
- C) *Walkthrough*.
- D) *Joint Application Development*.
- E) Análise Estruturada.

RESOLUÇÃO:

A Etnografia é um método utilizado pela antropologia na coleta de dados. Ela se baseia no contato entre o antropólogo e seu objeto de estudo, geralmente um grupo social constituído formalmente. Na engenharia de software a etnografia é caracterizada como uma técnica de observação utilizada para mapear requisitos implícitos que refletem processos reais dentro de um ambiente sistêmico. Compreender requisitos sociais e organizacionais, promover um entendimento dos aspectos culturais que regem o ambiente sistêmico e direcionam os procedimentos etnográficos. Para aplicar a referida técnica no processo de levantamento de requisitos é necessário estruturar um protocolo (conjunto de regras) etnográfico.

DOS PROTOCOLOS:

- 1) Identifique as áreas do negócio a ser observadas;
- 2) Identifique os usuários chaves de cada área;

- 3) Obtenha aprovação da gerência da empresa para aplicar a técnica e deixe clara a finalidade do estudo que será desenvolvido;
- 4) Insira o analista no ambiente de trabalho, **importante: não identifique o analista**, ele deve desempenhar algum papel (ou cargo) dentro do ambiente;
- 5) O analista deve colher informações sobre o cargo que desempenha e do restante do ambiente. Manuais, procedimentos, formulários, relatórios, estatísticas sobre a execução das tarefas e exceções devem ser colecionados;
- 6) O analista deve documentar as informações, utilize uma linguagem clara, concisa e consistente na documentação;
- 7) Consolida o estudo efetuado pelos analistas;
- 8) Valide as informações consolidadas.

Resposta: A

39.(FCC - 2010 - DPE-SP - Agente de Defensoria - 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 afirmar em:

- A) III, apenas.
- B) I e II, apenas.
- C) I e III, apenas.
- D) II e III, apenas.
- E) I, II e III.

RESOLUÇÃO:

O erro da questão está em: II – os requisitos de sistemas podem descrever apenas requisitos não funcionais, entretanto, sabemos que isto é uma inverdade, pois os requisitos podem descrever itens funcionais que podem ser a descrição e funções dos sistemas, como requisitos não funcionais no que envolva a segurança destes.

Resposta: C

40.(CESPE - 2015 - TRE-GO - Técnico Judiciário - Programação de Sistemas)

A respeito de desenvolvimento e manutenção de sistemas, julgue o item seguinte.

Requisitos não funcionais são as restrições sobre os serviços ou declarações de funções que o sistema deve fornecer.

() Certo.

() Errado.

RESOLUÇÃO:

Requisitos não funcionais: "Requisitos não funcionais são restrições sobre os serviços ou as funções oferecidas pelo sistema. São aqueles que não dizem respeito diretamente às funções específicas fornecidas pelo sistema. Entre eles, destacam-se restrições de tempo de desenvolvimento ou de resposta, restrições sobre o processo de desenvolvimento, padrões, entre outros" (Engenharia de Software, Ian Sommerville).

Resposta: Errado

41.(CESPE - 2015 - TCU - Auditor Federal de Controle Externo - Tecnologia da Informação)

Julgue o próximo item, relativo ao CMMI, ao MPS.BR e a engenharia de requisitos.

Em todos os sistemas novos, o processo de engenharia de requisitos deve começar por um estudo de viabilidade.

() Certo.

() Errado.

RESOLUÇÃO:

Isso mesmo meu amigo, tudo se inicia pelo estudo de viabilidades.

Mais o que é o estudo de viabilidade: O objetivo de um estudo de viabilidade, como o próprio nome sugere, é avaliar sob o ponto de vista operacional, técnico, econômico e organizacional se o projeto é viável.

O documento destina-se ao *stakeholders* do sistema (alguém que tenha alguma influência direta ou indireta sobre os requisitos do sistema). (SOMMERVILLE, 2011, p. 70).

O estudo de viabilidade acontece após a especificação de requisitos de negócio, ou seja, é o segundo passo do processo de engenharia de requisitos.

Resposta: Certo

42.(CESPE - 2015 - STJ - Técnico Judiciário - Tecnologia da Informação)

Acerca de processos e práticas ágeis de desenvolvimento e de análise de requisitos funcionais e não funcionais, julgue o item a seguir.

Serão classificados como organizacionais os requisitos operacionais e de desenvolvimento que forem identificados na análise de requisitos não funcionais.

() Certo.

() Errado.

RESOLUÇÃO:

A principal diferença entre funcional e o não funcional são: funcional, se preocupa como determinadas coisas no sistema funcionarão, dados que deverão existir nos cadastros entre outros, já no não funcional, este se preocupa principalmente com a questão de segurança.

Resposta: Certo

43.(CESPE - 2015 - STJ - Técnico Judiciário - Tecnologia da Informação)

Acerca de processos e práticas ágeis de desenvolvimento e de análise de requisitos funcionais e não funcionais, julgue o item a seguir.

Na análise de requisitos não funcionais, o tempo de treinamento, o tempo de reinício após falha, o número de sistemas-alvo e a disponibilidade são medidas para, respectivamente, facilidade de uso, robustez, portabilidade e confiabilidade.

() Certo.

() Errado.

RESOLUÇÃO:

Perfeito, afinal, a parte de segurança do sistema integra os itens do processo de requisitos não funcionais.

Resposta: Certo

44.(CESPE - 2015 - MPOG - Analista em Tecnologia da Informação - Cargo 12)

Julgue o item a seguir, a respeito do ciclo de vida e das metodologias de desenvolvimento de *software*.

No ciclo de vida do *software*, o congelamento dos requisitos do *software* garante que este, quando em desenvolvimento, atenda à expectativa do usuário, desde que tudo que tenha sido requisitado seja implementado.

- () Certo.
() Errado.

RESOLUÇÃO:

O que garante o sistema de acordo com as expectativas do cliente é o sistema entregue com todas as funcionalidades e recursos solicitados e levantados através de uma análise de requisitos séria e competente.

Quanto ao congelamento de requisito, na verdade eles são estáticos. Imagine um item de requisito congelado e outros itens que dependem dele dinâmicos e ao mudar uma regra, como ficaria? Seria um desastre de proporções gigantescas.

Resposta: Errado

45.(CESPE - 2018 - FUB - Técnico de Tecnologia da Informação)

Julgue o item que se segue, relativo a disciplinas do processo de desenvolvimento de *software*.

O documento de requisitos deve ser elaborado a partir da análise de viabilidade do *software*, seguida de análise, especificação e validação de requisitos.

- () Certo.
() Errado.

RESOLUÇÃO:

Certíssimo, tudo começa com análise de viabilidade, e nela se elabora o documento, o qual será o início de todo o processo.

Resposta: Certo

46.(CESPE - 2014 - ANATEL - Técnico em Regulação - Telecomunicações)

Em relação à elicitação de requisitos, julgue o item subsequente.

A análise de documentos, forma de elicitação de requisitos realizada pela identificação de informações relevantes e pelo estudo de documentações, abrange, entre outros elementos, estudos de mercado, contratos e requisições de propostas.

- () Certo.
() Errado.

RESOLUÇÃO:

A análise de documentos abrange toda a área da análise de requisitos.

Resposta: Certo

47.(CESPE - 2014 - ANATEL - Técnico em Regulação - Telecomunicações)

Em relação à elicitação de requisitos, julgue o item subsequente.

Na técnica de análise de interface, os requisitos da interface do usuário são detalhados e integrados a outros requisitos, como casos de uso, cenários, regras de dados e de negócio, o que permite ao especificador de requisitos avaliar, identificar, descrever e validar suas necessidades de interface da solução a ser desenvolvida.

() Certo.

() Errado.

RESOLUÇÃO:

Amigos, aqui outra questãozinha polêmica da Banca Examinadora, eles simplesmente trocaram o nome da técnica.

Resposta: Errado

48.(CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

Com relação à engenharia de software, julgue o próximo item.

Na engenharia de *software*, a etapa de análise de requisitos pode ser dividida nas seguintes áreas: reconhecimento do problema; avaliação e síntese; modelagem; especificação; e revisão.

() Certo.

() Errado.

RESOLUÇÃO:

Correto, a análise de requisitos de software pode ser dividida em cinco áreas de esforço: reconhecimento do problema, avaliação e síntese, modelagem, especificação e revisão. (PRESSMAN, 2002).

Resposta: Certo

49.(CESPE - 2016 - TCE-PA - Auditor de Controle Externo - Área Informática - Web Design)

Julgue o próximo item acerca de análise de requisitos de usabilidade, teste de usabilidade e de interfaces *web*.

Os requisitos de usabilidade devem ser estabelecidos em função do conteúdo, dos usuários e do contexto de uso.

- () Certo.
() Errado.

RESOLUÇÃO:

A usabilidade advém de um cenário onde é observado com técnicas o uso do sistema, interação do usuário final, neste processo podemos analisar o nível também de aceitação da ferramenta pelo usuário.

Resposta: Certo

50.(CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsecutivo, acerca de análise de requisitos.

A prototipação deve ser realizada na fase final de um projeto, no momento em que o programa passou da fase de testes finais e já se encontra devidamente em produção.

- () Certo.
() Errado.

RESOLUÇÃO:

Muito pelo contrário, a prototipação deve ser realizada no início do processo. Mas o que vem a ser prototipação? É criar modelos protótipo do sistema com telas parecidas com o que será construído.

Resposta: Errado

51.(CESPE - 2018 - STJ - Técnico Judiciário - Desenvolvimento de Sistemas)

Julgue o próximo item, a respeito de engenharia de *software* e análise de requisitos.

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

- () Certo.
() Errado.

RESOLUÇÃO:

Errado, algumas das características citadas acima são de requisitos não funcionais a exemplo: segurança, usabilidade, escalabilidade, tempo de resposta.

Resposta: Errado

52.(CESPE - 2015 - MEC - Analista de Processos)

No que se refere à análise de negócios, julgue o item a seguir.

O conceito de requisito está limitado a condições ou capacidades passadas, presentes e futuras em uma organização, e descreve estruturas organizacionais, papéis, processos, políticas, regras e sistemas de informação.

() Certo.

() Errado.

RESOLUÇÃO:

Muito pelo contrário, o termo requisito é usado de maneira bem mais ampla, se você parar para pensar, requisito envolve toda e qualquer função do sistema e suas características.

Resposta: Errado

53.(CESPE - 2016 - FUB - Técnico de Tecnologia da Informação)

Acerca dos conceitos de análise e projeto de sistemas em engenharia de *software*, julgue o item subsequente.

O objetivo da tarefa de validação, realizada na etapa de análise de requisitos, consiste em assegurar que o *software* atenderá às necessidades levantadas pelo cliente.

() Certo.

() Errado.

RESOLUÇÃO:

Assertiva correta, a validação é a maneira de passar por todo o processo conferindo tudo o que foi solicitado e desenvolvido, desta maneira, visa a certeza de atender às necessidades do cliente.

Resposta: Certo

54.(CESPE - 2018 - STJ - Técnico Judiciário - Desenvolvimento de Sistemas)

Julgue o próximo item, a respeito de engenharia de *software* e análise de requisitos. *Software* é um conjunto de instruções (algoritmos) que, quando executadas, fornecem as características, as funções e o desempenho desejados para um sistema, para um programa ou para uma aplicação.

() Certo.

() Errado.

RESOLUÇÃO:

Este é o conceito básico de software. Pessoal, questão meia fora de nosso escopo, porém, como estava no meio do conteúdo vamos fazê-la, não custa nada, rs....

Resposta: Certo

55.(CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsecutivo, acerca de análise de requisitos.

Laddering é um método de entrevistas estruturadas, utilizado para o levantamento de conhecimento de especialistas, e que consiste na criação, revisão e modificação da hierarquia de conhecimento dos especialistas geralmente na forma de diagramas hierárquicos

() Certo.

() Errado.

RESOLUÇÃO:

Assertiva correta, este método de entrevista é estruturado, ele é utilizado para o levantamento de conhecimento referente a análise de requisitos.

Resposta: Certo

56.(CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

No que se refere a engenharia de *software*, aos modelos de ciclo de vida e aos requisitos funcionais e não funcionais, julgue o item a seguir.

Os requisitos não funcionais utilizam, para especificação, métricas como velocidade, confiabilidade, tamanho, análise de requisitos e classificação dos requisitos.

() Certo.

() Errado.

RESOLUÇÃO:

Classificação dos requisitos cabe ao requisito funcional.

Resposta: Errado

57.(CESPE - 2016 - TCE-PA - Auditor de Controle Externo - Área Informática - Web Design)

Julgue o próximo item acerca de análise de requisitos de usabilidade, teste de usabilidade e de interfaces web.

Os testes de usabilidade auxiliam o desenvolvedor a identificar, entre outros aspectos, o grau de entendimento da interface pelo usuário e as deficiências de suporte tecnológico.

() Certo.

() Errado.

RESOLUÇÃO:

Certíssimo, podemos usar esta técnica com a combinação de outras, desta maneira podemos identificar aceitação e possíveis dificuldades do usuário final com o sistema desenvolvido.

Resposta: Certo

58.(CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsecutivo, acerca de análise de requisitos.

Etnografia é uma técnica de observação que pode ser utilizada para compreender os requisitos sociais e organizacionais, isto é, entender a política organizacional, bem como a cultura de trabalho, com objetivo de familiarizar-se com o sistema e sua história

() Certo.

() Errado.

RESOLUÇÃO:

Esta técnica foi dada com a origem de inserir um analista dentro da empresa e observar o uso de sistemas e ferramentas, é de grande importância que quando usada a técnica os funcionários da empresa não saibam que o analista está a observar o comportamento de todos.

Resposta: Certo

59.(CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

No que se refere a engenharia de *software*, aos modelos de ciclo de vida e aos requisitos funcionais e não funcionais, julgue o item a seguir.

A prototipagem, técnica que serve de complemento à análise de requisitos, é a construção de protótipos considerados um esboço de parte do sistema.

() Certo.

() Errado.

RESOLUÇÃO:

Vimos uma questão parecida acima, o protótipo é uma imitação, algo parecido com o sistema que será desenvolvido.

Resposta: Certo

60.(INSTITUTO AOCP - 2015 - EBSEH - Analista de Tecnologia da Informação - Processos (HE-UFPEL))

A Engenharia de Software realiza tarefas de modelagem que levam à especificação completa dos requisitos e à representação abrangente do projeto para o software a ser construído. Em relação à fase de análise de requisitos, considere as seguintes assertivas:

- I. A análise de requisitos exige que o desenvolvedor descarte noções preconcebidas da corretividade do software recém-desenvolvido.
- II. A análise de requisitos fornece ao projetista de software uma representação da informação, função e comportamento, que podem ser traduzidos para os projetos arquitetural, de interfaces e em nível de componentes.
- III. A análise de requisitos de um programa é a estrutura ou estruturas do sistema que abrange os componentes de software, as propriedades externamente visíveis desses componentes e as relações entre eles.

Qual(is) assertiva(s) está(ão) corretas em relação à análise de requisitos?

- A) Apenas I.
- B) Apenas II.
- C) Apenas III.
- D) Apenas I e II.
- E) Apenas I e III.

RESOLUÇÃO:

A análise de requisitos é crucial, importante, pois ela **lavar**á ao projetista do sistema uma representação de como deve ser o sistema.

Resposta: B

61.(FUNIVERSA - 2010 - SEPLAG-DF - Analista de Sistemas)

Análise de requisitos está associada ao processo de descobrir as operações que um sistema deve realizar, bem como as restrições acerca dessas operações. A respeito de análise de requisitos, assinale a alternativa correta.

- A) Os requisitos descrevem apenas o que tem de ser feito pelo sistema, e não como deve ser feito.

- B) Entre as técnicas de elicitación de requisitos, estão cenários, entrevistas e triangulações.
- C) Nas entrevistas fechadas, não há uma agenda ou questões pré-definidas.
- D) O custo para corrigir problemas com requisitos não depende de esses problemas ser ou não descobertos mais cedo ou mais tarde no processo de desenvolvimento do sistema.
- E) Os requisitos funcionais descrevem o que o sistema deve fazer.

RESOLUÇÃO:

É isso aí pessoal, os requisitos funcionais descrevem como será o sistema. Lembrem-se que um levantamento de requisitos mal feito nos trará complicações futura com o sistema.

Resposta: E

62.(FCC - 2019 - SEMEF Manaus - AM - Assistente Técnico de Tecnologia da Informação da Fazenda Municipal - Programador)

Ao fazer uso da engenharia de requisitos em projetos, deve-se analisar o processo de elicitación e análise de requisitos, o qual pode ser dividido nas seguintes atividades:

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

A ordem sequencial correta para a execução dessas atividades é:

- A) I, III, IV e II.
- B) II, IV, III e I.
- C) III, II, IV e I.
- D) IV, I, II e III.
- E) III, I, II e IV.

RESOLUÇÃO:

A análise segue a seguinte ordem: Classificação e Organização de Requisitos, Priorização e Negociação de Requisitos, Documentação de Requisitos.

Resposta: C

Lista de questões

1.(CESPE - 2011 - MEC - Gerente de Projetos)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A engenharia de *software*, disciplina relacionada aos aspectos da produção de *software*, abrange somente os processos técnicos do desenvolvimento de *software*.

- () Certo.
() Errado.

2.(CESPE - 2013 - TRT - 10ª REGIÃO (DF e TO) - Analista Judiciário - Tecnologia da Informação)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A engenharia de *software* engloba processos, métodos e ferramentas. Um de seus focos é a produção de *software* de alta qualidade a custos adequados.

- () Certo.
() Errado.

3.(CESPE - 2010 - TRE-BA - Analista Judiciário - Análise de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Entre os desafios enfrentados pela engenharia de *software* estão lidar com sistemas legados, atender à crescente diversidade e atender às exigências quanto a prazos de entrega reduzidos.

- () Certo.
() Errado.

4.(CESPE - 2010 - DETRAN-ES - Analista de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Segundo princípio da engenharia de *software*, os vários artefatos produzidos ao longo do seu ciclo de vida apresentam, de forma geral, nível de abstração cada vez menor

- () Certo.
() Errado.

5. (CESPE - 2010 - TRE-BA - Técnico Judiciário - Programação de Sistemas)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Em uma visão restritiva, muitas pessoas costumam associar o termo *software* aos programas de computador. *Software* não é apenas o programa, mas também todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente.

A respeito de engenharia de *software*, julgue os itens

A engenharia de *software* está relacionada com todos os aspectos da produção de *software*, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação. A engenharia de sistemas diz respeito aos aspectos do desenvolvimento e da evolução de sistemas complexos, nos quais o *software* desempenha um papel importante.

() Certo.

() Errado.

6. (CESPE - 2013 - TRT - 10ª REGIÃO (DF e TO) - Analista Judiciário - Tecnologia da Informação)

Com relação a conceitos gerais da engenharia de *software*, julgue.

As atividades fundamentais relacionadas ao processo de construção de um *software* incluem a especificação, o desenvolvimento, a validação e a evolução do *software*.

() Certo.

() Errado.

7. (CESPE - 2013 - MPOG - Técnico de Nível Superior - V - Categoria Profissional 7)

Com relação a conceitos gerais da engenharia de *software*, julgue.

A evolução de um *software* é realizada ao ser testados, integrados e aceitos os componentes individuais.

() Certo.

() Errado.

8. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Engenharia de software não está relacionada somente aos processos técnicos de desenvolvimento de softwares, mas também a atividades como gerenciamento de projeto e desenvolvimento de ferramentas, métodos e teorias que apoiem a produção de softwares.

() Certo.

() Errado.

9. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação a conceitos gerais da engenharia de *software*, julgue.

Sistemas que incluem softwares são classificados em duas categorias: sistemas técnicos embasados em computadores e sistemas sócio técnicos. Os primeiros incluem componentes de hardware, software, pessoas, procedimentos e processos; os segundos são regidos pelas políticas e regras organizacionais.

() Certo.

() Errado.

10. (CESPE - 2013 - STF - Analista Judiciário - Análise de Sistemas de Informação)

Com referência a aspectos diversos de engenharia de software, julgue os itens subsecutivos.

Engenharia de *software* é um ramo da ciência da computação que se preocupa com o desenvolvimento de códigos que sejam confiáveis, seguros, baratos e que, quando executados, funcionem e tenham o desempenho desejado.

() Certo.

() Errado.

11. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Com referência a aspectos diversos de engenharia de software, julgue os itens subsecutivos.

O ciclo de vida clássico da engenharia de software tem sofrido críticas dos especialistas. Uma delas é o fato de o cliente ter de declarar todos os requisitos explicitamente na primeira parte do projeto, gerando insegurança. Uma técnica utilizada para minimizar esse problema é conhecida como:

- A) Análise por ponto de função.
 - B) Casos de teste.
 - C) Linha básica ou baseline.
 - D) Projeto de software.
 - E) Prototipação.
-

12. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

A análise por pontos de função utiliza diversas características para estimar o tamanho de um software. Das características abaixo, indique a que **NÃO** afeta a contagem nesse tipo de métrica.

- A) Desempenho.
 - B) Necessidade de backup.
 - C) Necessidade de testes.
 - D) Necessidade de comunicação de dados.
 - E) Número de entradas do usuário.
-

13. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Devido ao aumento da demanda por aplicações que funcionem 24 horas por dia na Internet, um software deve ser capaz de manter-se em operação após uma determinada falha. A estratégia de teste que melhor garante essa característica é o(a):

- A) Teste de estresse.
 - B) Teste de recuperação.
 - C) Teste unitário.
 - D) Debugging.
 - E) Simulação combinatória.
-

14. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Na análise estruturada, as duas ferramentas de modelagem que identificam de modo mais adequado a estrutura dos dados são:

- A) Diagrama de entidades-relacionamentos e dicionário de dados.
- B) Diagrama de transições de estado e diagrama de fluxo de dados.
- C) Diagrama de fluxo de dados e dicionário de dados.
- D) Diagrama de fluxo de dados e diagrama de entidades relacionamentos.
- E) Dicionário de dados e diagrama de transições de estado.

15. (CESGRANRIO - 2007 - EPE - Analista de Gestão Corporativa Júnior - Área Tecnologia da Informação)

Um processo administrativo numa repartição pública poderá estar nos seguintes estados: aberto, na carga de um determinado funcionário, em trâmite ou arquivado. A ferramenta de modelagem que representa de modo mais adequado as regras de passagem de um estado para outro é:

- A) Diagrama de fluxo de dados.
- B) Diagrama de transições de estado.
- C) Diagrama entidades-relacionamentos.
- D) Dicionário de dados.
- E) Especificação de processos.

16. (CESPE - 2010 - SERPRO - Analista Desenvolvimento de Sistemas | CESPE - 2010 - SERPRO - Analista - Desenvolvimento de Sistemas)

A engenharia de software é uma disciplina do conhecimento humano que aplica princípios da engenharia ao desenvolvimento de software. No entanto, existem diferenças significativas entre as engenharias clássicas — mecânica, civil, elétrica — e a engenharia de software, muitas delas decorrentes das diferenças de natureza entre o produto ou sistema resultante da atividade das engenharias clássicas e a engenharia de software. A engenharia de software desdobrou-se em várias áreas especializadas, como as áreas de requisitos de software, de análise e projeto de software, de implementação de software, de testes de software, de gestão de configuração, entre outras. Na interface entre as engenharias clássicas e a de software insere-se a engenharia de sistemas, abordada colateralmente na ISO/IEC 12207.

A homologação de um *software* é uma atividade pertinente aos processos de aquisição e não aos processos de garantia de qualidade, sendo ainda diretamente relacionada ao ciclo de vida de sistemas e não apenas de *software*.

() Certo.

() Errado.

17. (CESPE - 2013 - TCE-RO - Analista de Informática)

Com relação à engenharia de software, julgue os itens seguintes.

A engenharia de software define quatro fases para o ciclo de vida de um sistema: iniciação, elaboração, construção e transição, sendo essa última responsável pela homologação dos artefatos junto ao cliente.

() Certo.

() Errado.

18. (CESPE - 2010 - TRE-BA - Técnico Judiciário - Programação de Sistemas)

Em uma visão restritiva, muitas pessoas costumam associar o termo *software* aos programas de computador. *Software* não é apenas o programa, mas também todos os dados de documentação e configuração associados, necessários para que o programa opere corretamente. A respeito de engenharia de *software*, julgue os itens.

A engenharia de *software* está relacionada com todos os aspectos da produção de *software*, desde os estágios iniciais de especificação do sistema até sua manutenção, depois que este entrar em operação. A engenharia de sistemas diz respeito aos aspectos do desenvolvimento e da evolução de sistemas complexos, nos quais o *software* desempenha um papel importante.

() Certo.

() Errado.

19. (CESPE - 2008 - INPE - Tecnologista Júnior - TS12)

Com relação aos conceitos de engenharia de software, julgue os próximos itens.

A técnica de inspeção de software é mais custosa do que os testes.

() Certo.

() Errado.

20. (CESPE - 2008 - INPE - Tecnologista Júnior - TS12)

Com relação aos conceitos de engenharia de software, julgue os próximos itens.

As abstrações de dados estão entre os tipos de módulos que podem ser criados durante o processo de modularização de um programa.

- () Certo.
() Errado.

21. (CESPE - 2014 - TC-DF - Analista de Administração Pública - Sistemas de TI)

Julgue os itens a seguir, com relação à engenharia de software.

O modelo de negócio é definido como um desenho das atividades sequenciadas geradas por entradas e que geram também saídas, apoiadas por artefatos específicos

- () Certo.
() Errado.

22. (CESPE - 2014 - ANATEL - Analista Administrativo - Desenvolvimento de Sistemas)

A respeito de engenharia de *software*, julgue os próximos itens.

Em sistemas em tempo real orientados a eventos, é adequada a utilização de um modelo de controle orientado à interrupção.

- () Certo.
() Errado.

23. (CESPE - 2014 - ANATEL - Analista Administrativo - Desenvolvimento de Sistemas)

A respeito de engenharia de *software*, julgue os próximos itens.

A recursividade é uma técnica que pode ser utilizada na implementação de sistemas de lógica complexa, com a finalidade de minimizar riscos de ocorrência de defeitos no *software*.

- () Certo.
() Errado.

24. (CESPE - 2013 - BACEN - Analista - Análise e Desenvolvimento de Sistemas)

A engenharia de *software* engloba, além dos processos técnicos de desenvolvimento de *software*, atividades de gerenciamento de projeto de *software* e desenvolvimento de ferramentas, métodos e procedimento para apoiar a produção de *software*. Oferece uma base para a construção de *software* de alta qualidade e produtividade. Com relação à engenharia de *software*, julgue o item que se segue.

Procedimento é o conjunto de tarefas que incluem os planejamentos e estimativas de projeto, análise de requisitos de sistema, modelagem de dados, arquitetura, codificação, teste e manutenção.

() Certo.

() Errado.

25. (CESPE - 2013 - BACEN - Analista - Análise e Desenvolvimento de Sistemas)

Julgue o item a seguir, relativo a análise e projeto de *software*.

A linguagem ubíqua utiliza termos que fazem parte das discussões entre os especialistas de negócio e as equipes de desenvolvimento, os quais devem utilizar a mesma terminologia na linguagem falada e no código.

() Certo.

() Errado.

26. (CESPE - 2013 - SEGESP-AL - Perito Criminal - Tecnologia da Informação)

Em relação aos conceitos de engenharia de *software* e ciclo de vida de um *software*, julgue o item a seguir.

O desenvolvimento de produto de *software* por meio da utilização da técnica de prototipação consiste na execução das atividades de desenvolvimento de *software* em uma sequência ordenada, sendo as principais atividades do modelo requisitos de sistema; requisitos de *software*; análise; projeto de programa; codificação; teste; e operação.

() Certo.

() Errado.

27. (FCC - 2015 - TRE-AP - Técnico Judiciário - Programação de Sistemas)

Um Técnico participou do levantamento de requisitos de um novo sistema do Tribunal. Devidamente autorizado, ele se inseriu no ambiente de trabalho em que o sistema seria utilizado e observou o trabalho diário, anotando as tarefas reais. Seu principal objetivo era descobrir requisitos de sistema implícitos, que refletissem

os processos reais nos quais as pessoas estão envolvidas, ao invés de processos formais. Além destes requisitos, ele também coletou os requisitos derivados da cooperação e conscientização das atividades de outras pessoas envolvidas. O Técnico estava colocando em prática a técnica de levantamento de requisitos denominada.

- A) VORD (*Viewpoint-Oriented Requirements Definition*) ou Definição de Requisitos Orientada a Ponto de Vista.
- B) Etnografia.
- C) JAD (*Joint Application Design*).
- D) *Workshop* ou Oficinas de Trabalho.
- E) Prototipagem.

28.(CESGRANRIO - 2012 - LIQUIGÁS - Profissional Júnior - Administração de Banco de Dados)

Ao levantar os requisitos necessários para a implantação de um workflow em uma empresa, a equipe responsável percebeu que o requisito mais importante era a facilidade de definir o processo, sendo a flexibilidade mais relevante que a produtividade. Também era desejado que as definições de processos fossem criadas com formulários.

Nesse caso, o tipo de workflow mais adequado é o:

- A) Ad hoc.
- B) Administrativo.
- C) Autônomo.
- D) Colaborativo.
- E) Produção.

29.(CESGRANRIO - 2006 - Petrobrás - Analista de Sistemas Pleno - Engenharia de Software | CESGRANRIO - 2006 - Petrobrás - Analista de Sistemas Pleno - Processos)

Sobre a Análise e o Gerenciamento de Requisitos, é **FALSO** afirmar que:

- A) Quanto mais tarde for identificado um problema na análise de requisitos, maior será o custo com o retrabalho.
- B) A elicitación é o processo de identificação e entendimento das necessidades e restrições dos usuários, enquanto que a especificação é o processo de formalização das necessidades e restrições dos usuários em requisitos funcionais de software.

- C) Na análise de requisitos o cliente utiliza as melhores práticas de engenharia de requisitos na tarefa de descrever suas necessidades.
- D) O gerenciamento de requisitos corresponde ao conjunto de atividades que auxilia a equipe do projeto a identificar, controlar e rastrear os requisitos, bem como a fazer as alterações nos requisitos durante o projeto.
- E) O gerenciamento de requisitos implica a alteração, inclusão e/ou exclusão de requisitos ao produto de software, o que pode levar a alterações de prazos, de recursos humanos, de equipamentos e de tecnologia.

30.(CESPE - 2004 - Polícia Federal - Perito Criminal Federal - Informática)

Considere que se deseja desenvolver um sistema para controle de caixa de supermercado tendo como base um computador que registra os produtos vendidos, interagindo com dispositivos de entrada e saída tais como impressora, teclado e leitora de código de barras. Esse sistema deve interagir também com o operador do caixa e com um banco de dados do estabelecimento. A partir dessas informações, julgue os itens que se seguem.

A descrição informal do que o sistema deve fazer, tal como ler código de barras, identificar o produto e calcular o total da compra, faz parte da especificação de requisitos do programa.

- () Certo.
- () Errado.

31.(CESGRANRIO - 2008 - BR Distribuidora - Analista de Sistemas Júnior - Processos de Negócios)

Analise as afirmativas a seguir, sobre requisitos em projetos de software.

- I - O rastreamento de requisitos é de grande importância para conduzir análises de impacto quando há mudanças em requisitos.
- II - O acrônimo FURPS+ se refere aos requisitos não funcionais das categorias de Feasibility, Usability, Reliability, Performance e Supportability.
- III - Um requisito pode conter, além da especificação, atributos que sirvam ao seu gerenciamento.
- IV - Casos de uso são descrições da interação entre um ator e o sistema e, portanto, especificam apenas requisitos funcionais.

Estão corretas **APENAS** as afirmativas

- A) I e II.

- B) I e III.
- C) II e III.
- D) II e IV.
- E) III e IV.

32. (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.

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

Com relação aos requisitos de software, considere:

- IV. Funcionais são somente requisitos de usuário.
- V. Funcionais e não-funcionais podem ser requisitos de usuário.
- VI. Funcionais e não-funcionais podem ser requisitos de sistema.

Está correto o que se afirmar APENAS em:

- A) I.
- B) II.
- C) III.
- D) I e III.
- E) II e III.

34.(FCC - 2008 - TRT - 18ª Região (GO) - Analista Judiciário - Tecnologia da Informação)

Considerando que a especificação dos requisitos pode não ser completa durante o estágio de Análise de Requisitos, em razão da imaturidade de conhecimento de clientes e desenvolvedores, é recomendável que a análise e modelagem dos requisitos tenham uma abordagem.

- A) operacional, somente.
- B) bottom-up.
- C) conceitual, somente.
- D) iterativa.
- E) estratégica, somente.

35.(FCC - 2010 - DPE-SP - Agente de Defensoria - Programador)

Na Engenharia de Requisitos, dentre passos a serem seguidos para elicitação de requisitos é INCORRETO: operacional, somente.

- A) Solicitar participação de várias pessoas para que os requisitos sejam definidos a partir de diversos pontos de vista.
- B) Identificar regras de domínio que limitam a funcionalidade ou desempenho do sistema ou produto que será construído.
- C) Definir um ou mais métodos de elicitação de requisitos.
- D) Selecionar as pessoas sem preconceitos organizacionais para auxiliar a especificar os requisitos.
- E) Identificar claramente a justificativa de existência para cada requisito registrado; identificar requisitos ambíguos que serão candidatos a prototipação.

36.(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.

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

No âmbito da Engenharia de *Software*, um cenário de constantes mudanças políticas como as que ocorrem em uma aplicação governamental, por exemplo, propõe a especificação de um elemento de *software* que é o:

- A) Requisito inconsistente.
- B) Requisito volátil.
- C) Processo indefinido
- D) Escopo reverso.
- E) Escopo inconsistente.

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

No processo de elicitação e análise de requisitos, a técnica pela qual o analista, como recurso, insere-se no ambiente de trabalho onde o sistema será usado, observando e registrando a rotina diária dos envolvidos para descobrir requisitos implícitos do sistema que reflitam os processos reais e não os formais, denomina-se:

- A) Etnografia.
- B) Análise de Dados.
- C) *Walkthrough*.
- D) *Joint Application Development*.
- E) Análise Estruturada.

39.(FCC - 2010 - DPE-SP - Agente de Defensoria - Analista de Sistemas)

Sobre análise de requisitos da engenharia de software, considere:

- II. 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 afirmar em:

- A) III, apenas.
- B) I e II, apenas.

- C) I e III, apenas.
- D) II e III, apenas.
- F) I, II e III.

40.(CESPE - 2015 - TRE-GO - Técnico Judiciário - Programação de Sistemas)

A respeito de desenvolvimento e manutenção de sistemas, julgue o item seguinte.

Requisitos não funcionais são as restrições sobre os serviços ou declarações de funções que o sistema deve fornecer.

- () Certo.
- () Errado.

41.(CESPE - 2015 - TCU - Auditor Federal de Controle Externo - Tecnologia da Informação)

Julgue o próximo item, relativo ao CMMI, ao MPS.BR e a engenharia de requisitos.

Em todos os sistemas novos, o processo de engenharia de requisitos deve começar por um estudo de viabilidade.

- () Certo.
- () Errado.

42.(CESPE - 2015 - STJ - Técnico Judiciário - Tecnologia da Informação)

Acerca de processos e práticas ágeis de desenvolvimento e de análise de requisitos funcionais e não funcionais, julgue o item a seguir.

Serão classificados como organizacionais os requisitos operacionais e de desenvolvimento que forem identificados na análise de requisitos não funcionais.

- () Certo.
- () Errado.

43.(CESPE - 2015 - STJ - Técnico Judiciário - Tecnologia da Informação)

Acerca de processos e práticas ágeis de desenvolvimento e de análise de requisitos funcionais e não funcionais, julgue o item a seguir.

Na análise de requisitos não funcionais, o tempo de treinamento, o tempo de reinício após falha, o número de sistemas-alvo e a disponibilidade são medidas para, respectivamente, facilidade de uso, robustez, portabilidade e confiabilidade.

() Certo.

() Errado.

44.(CESPE - 2015 - MPOG - Analista em Tecnologia da Informação - Cargo 12)

Julgue o item a seguir, a respeito do ciclo de vida e das metodologias de desenvolvimento de *software*.

No ciclo de vida do *software*, o congelamento dos requisitos do *software* garante que este, quando em desenvolvimento, atenda à expectativa do usuário, desde que tudo que tenha sido requisitado seja implementado.

() Certo.

() Errado.

45.(CESPE - 2018 - FUB - Técnico de Tecnologia da Informação)

Julgue o item que se segue, relativo a disciplinas do processo de desenvolvimento de *software*.

O documento de requisitos deve ser elaborado a partir da análise de viabilidade do *software*, seguida de análise, especificação e validação de requisitos.

() Certo.

() Errado.

46.(CESPE - 2014 - ANATEL - Técnico em Regulação - Telecomunicações)

Em relação à elicitação de requisitos, julgue o item subsequente.

A análise de documentos, forma de elicitação de requisitos realizada pela identificação de informações relevantes e pelo estudo de documentações, abrange, entre outros elementos, estudos de mercado, contratos e requisições de propostas.

- () Certo.
() Errado.
-

47.(CESPE - 2014 - ANATEL - Técnico em Regulação - Telecomunicações)

Em relação à elicitación de requisitos, julgue o item subsequente.

Na técnica de análise de interface, os requisitos da interface do usuário são detalhados e integrados a outros requisitos, como casos de uso, cenários, regras de dados e de negócio, o que permite ao especificador de requisitos avaliar, identificar, descrever e validar suas necessidades de interface da solução a ser desenvolvida.

- () Certo.
() Errado.
-

48.(CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

Com relação à engenharia de software, julgue o próximo item.

Na engenharia de *software*, a etapa de análise de requisitos pode ser dividida nas seguintes áreas: reconhecimento do problema; avaliação e síntese; modelagem; especificação; e revisão.

- () Certo.
() Errado.
-

49.(CESPE - 2016 - TCE-PA - Auditor de Controle Externo - Área Informática - Web Design)

Julgue o próximo item acerca de análise de requisitos de usabilidade, teste de usabilidade e de interfaces *web*.

Os requisitos de usabilidade devem ser estabelecidos em função do conteúdo, dos usuários e do contexto de uso.

- () Certo.
() Errado.
-

50.(CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsecutivo, acerca de análise de requisitos.

A prototipação deve ser realizada na fase final de um projeto, no momento em que o programa passou da fase de testes finais e já se encontra devidamente em produção.

- () Certo.
() Errado.

51.(CESPE - 2018 - STJ - Técnico Judiciário - Desenvolvimento de Sistemas)

Julgue o próximo item, a respeito de engenharia de *software* e análise de requisitos.

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

- () Certo.
() Errado.

52.(CESPE - 2015 - MEC - Analista de Processos)

No que se refere à análise de negócios, julgue o item a seguir.

O conceito de requisito está limitado a condições ou capacidades passadas, presentes e futuras em uma organização, e descreve estruturas organizacionais, papéis, processos, políticas, regras e sistemas de informação.

- () Certo.
() Errado.

53.(CESPE - 2016 - FUB - Técnico de Tecnologia da Informação)

Acerca dos conceitos de análise e projeto de sistemas em engenharia de *software*, julgue o item subsequente.

O objetivo da tarefa de validação, realizada na etapa de análise de requisitos, consiste em assegurar que o *software* atenderá às necessidades levantadas pelo cliente.

- () Certo.
() Errado.

54. (CESPE - 2018 - STJ - Técnico Judiciário - Desenvolvimento de Sistemas)

Julgue o próximo item, a respeito de engenharia de *software* e análise de requisitos. *Software* é um conjunto de instruções (algoritmos) que, quando executadas, fornecem as características, as funções e o desempenho desejados para um sistema, para um programa ou para uma aplicação.

- () Certo.
- () Errado.
-

55. (CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsequente, acerca de análise de requisitos.

Laddering é um método de entrevistas estruturadas, utilizado para o levantamento de conhecimento de especialistas, e que consiste na criação, revisão e modificação da hierarquia de conhecimento dos especialistas geralmente na forma de diagramas hierárquicos

- () Certo.
- () Errado.
-

56. (CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

No que se refere a engenharia de *software*, aos modelos de ciclo de vida e aos requisitos funcionais e não funcionais, julgue o item a seguir.

Os requisitos não funcionais utilizam, para especificação, métricas como velocidade, confiabilidade, tamanho, análise de requisitos e classificação dos requisitos.

- () Certo.
- () Errado.
-

57.(CESPE - 2016 - TCE-PA - Auditor de Controle Externo - Área Informática - Web Design)

Julgue o próximo item acerca de análise de requisitos de usabilidade, teste de usabilidade e de interfaces web.

Os testes de usabilidade auxiliam o desenvolvedor a identificar, entre outros aspectos, o grau de entendimento da interface pelo usuário e as deficiências de suporte tecnológico.

() Certo.

() Errado.

58.(CESPE - 2015 - MEC - Analista de Sistemas)

Julgue o item subsequente, acerca de análise de requisitos.

Etnografia é uma técnica de observação que pode ser utilizada para compreender os requisitos sociais e organizacionais, isto é, entender a política organizacional, bem como a cultura de trabalho, com objetivo de familiarizar-se com o sistema e sua história

() Certo.

() Errado.

59.(CESPE - 2015 - TCE-RN - Assessor Técnico de Informática - Cargo 1)

No que se refere a engenharia de *software*, aos modelos de ciclo de vida e aos requisitos funcionais e não funcionais, julgue o item a seguir.

A prototipagem, técnica que serve de complemento à análise de requisitos, é a construção de protótipos considerados um esboço de parte do sistema.

() Certo.

() Errado.

60.(INSTITUTO AOCP - 2015 - EBSEH - Analista de Tecnologia da Informação - Processos (HE-UFPEL))

A Engenharia de Software realiza tarefas de modelagem que levam à especificação completa dos requisitos e à representação abrangente do projeto para o software a ser construído. Em relação à fase de análise de requisitos, considere as seguintes assertivas:

I. A análise de requisitos exige que o desenvolvedor descarte noções preconcebidas da corretividade do software recém-desenvolvido.

II. A análise de requisitos fornece ao projetista de software uma representação da informação, função e comportamento, que podem ser traduzidos para os projetos arquitetural, de interfaces e em nível de componentes.

III. A análise de requisitos de um programa é a estrutura ou estruturas do sistema que abrange os componentes de software, as propriedades externamente visíveis desses componentes e as relações entre eles.

Qual(is) assertiva(s) está(ão) corretas em relação à análise de requisitos?

- A) Apenas I.
- B) Apenas II.
- C) Apenas III.
- D) Apenas I e II.
- E) Apenas I e III.

61.(FUNIVERSA - 2010 - SEPLAG-DF - Analista de Sistemas)

Análise de requisitos está associada ao processo de descobrir as operações que um sistema deve realizar, bem como as restrições acerca dessas operações. A respeito de análise de requisitos, assinale a alternativa correta.

- A) Os requisitos descrevem apenas o que tem de ser feito pelo sistema, e não como deve ser feito.
- B) Entre as técnicas de elicitação de requisitos, estão cenários, entrevistas e triangulações.
- C) Nas entrevistas fechadas, não há uma agenda ou questões pré-definidas.
- D) O custo para corrigir problemas com requisitos não depende de esses problemas ser ou não descobertos mais cedo ou mais tarde no processo de desenvolvimento do sistema.
- E) Os requisitos funcionais descrevem o que o sistema deve fazer.

62.(FCC - 2019 - SEMEF Manaus - AM - Assistente Técnico de Tecnologia da Informação da Fazenda Municipal - Programador)

Ao fazer uso da engenharia de requisitos em projetos, deve-se analisar o processo de elicitação e análise de requisitos, o qual pode ser dividido nas seguintes atividades:

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

A ordem sequencial correta para a execução dessas atividades é:

- A) I, III, IV e II.
- B) II, IV, III e I.

- C) III, II, IV e I.
 - D) IV, I, II e III.
 - E) III, I, II e IV.
-

Gabarito

1. E	22. C	43. C
2. C	23. E	44. E
3. C	24. E	45. C
4. C	25. C	46. C
5. C	26. E	47. E
6. C	27. B	48. C
7. E	28. B	49. C
8. C	29. C	50. E
9. E	30. C	51. E
10. E	31. B	52. E
11. E	32. D	53. C
12. C	33. E	54. C
13. C	34. D	55. C
14. A	35. D	56. E
15. B	36. E	57. C
16. C	37. B	58. C
17. E	38. A	59. C
18. C	39. C	60. B
19. E	40. E	61. E
20. C	41. C	62. C
21. E	42. C	

Resumo direcionado

Conceitos básicos

Dos conceitos da Engenharia de Softwares, temos:

“Engenharia de software é o estabelecimento e o emprego de sólidos princípios de engenharia de modo a obter software de maneira econômica, que seja confiável e funcione de forma eficiente em máquinas reais”.

“Engenharia de *software* é uma área da computação voltada à especificação, desenvolvimento, manutenção e criação de *software*, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando a organização, produtividade e qualidade”.

Pessoal, com o fundamento nos dois conceitos descritos acima, sobre o que vem a ser a Engenharia de Software, já conseguiremos desenhar um pouco do que é esta disciplina e seus fundamentos e como são empregados em nosso dia a dia. Atualmente, as tecnologias e práticas da Engenharia de Software, englobam as mais diversas linguagens de programação, banco de dados, ferramentas e entre elas as IDEs, plataformas, bibliotecas, padrões de projeto de software, processo de software e qualidade de software. Entretanto, a engenharia de software deve oferecer mecanismos para que possamos planejar e gerenciar o processo de desenvolvimento de um sistema computacional de qualidade e que atenda às necessidades de um requisitante de software.

As bases científicas para a Engenharia de Software se relacionam com o uso de modelos abstratos que permitem ao engenheiro especificar, projetar, implementar e manter sistemas de software, avaliando e garantindo suas qualidades. A área responsável que estuda e avalia os processos da engenharia de software, propondo a evolução dos processos, ferramentas e métodos de suporte a engenharia.

Do Histórico

O termo Engenharia de Software foi criado na década de 1960, porém só utilizado oficialmente por volta do ano 1968, na NATO Science Commite. Sua criação, emergiu na tentativa de contornar a crise do software e dar um tratamento de engenharia mais sistemático, controlado e qualidade mensurável ao desenvolvimento de sistemas e software complexos.

Um sistema de software complexo caracteriza-se por um conjunto de componentes abstratos de software, estruturas de dados e algoritmos encapsulados na forma de algoritmos, funções, módulos, objetos ou agentes interconectados, compondo a arquitetura do software, que deverão ser executados em sistemas computacionais.

No início, e ainda hoje em algumas empresas menores com a falta de profissionais experientes, desenvolver um sistema é uma jornada quase que infinita, ao oposto disso, a Engenharia veio para: criar modelos, padrões, processos, metodologias, documentações, uma linha de desenvolvimento. Pois como narrei no período inicial deste parágrafo, este processo era uma aventura e que na maioria das vezes a empresa a qual pagava pela construção de um determinado software sempre ficava no prejuízo e as razões eram as mais diversas, podemos

até citar algumas destas: tempo prolongado no desenvolvimento, linguagem de programação não adequada, a não realização de levantamentos de requisitos funcionais e não funcionais, antes de entrar em produção o software já não atendia aos requisitos básicos ao qual foi construído, entre centenas de problemas que poderiam causar o seu insucesso.

A figura abaixo nos dá uma definição básica de fases da construção de um software.



Quando usamos das ferramentas certas ou adequadas, os métodos eficientes, o processo documentado e seguindo um fluxo, a qualidade se torna o resumo desta abstração.

Definição de Software

Amigos, começamos falando da Engenharia de Software, mas o que é realmente um software?

“É uma sequência de instruções a ser seguidas e/ou executadas, na manipulação, redirecionamento ou modificação de um dado (informação) ou acontecimento”.

“Software, também é o nome dado ao comportamento exibido por essa sequência de instruções, quando executada em um computador ou máquina semelhante, além de um produto desenvolvido pela engenharia de software, e inclui não só o programa de computador propriamente dito, mas também manuais e especificações”.

A respeito do software, para fins contábeis e financeiros, ele é considerado um bem de consumo!

“Um software normalmente é composto por diversas funções, bibliotecas e módulos que gera um programa executável ao final do processo de desenvolvimento e este, quando executado, recebe algum tipo de entrada de dados, no caso (*input*), processa as informações segundo uma série de algoritmos ou sequências de instruções lógicas e libera uma saída, esta, o (*output*) como resultado deste processamento. Um software bem desenvolvido é normalmente criado pela área da engenharia de software e inclui não apenas o programa de computador em si, mas também manuais, especificações e configurações”.

Bom pessoal, agora que sabemos na prática o que é um software, vamos partir para suas aplicações.

Campos de Aplicação

Os campos de aplicações são os mais diversos, pois a Engenharia de Software surgiu para colocar ordem na casa, rs. Antigamente, se desenvolvia um projeto de qualquer maneira, as datas de entrega da ferramenta quase sempre eram desobedecidas, construía-se algo de qualquer jeito, o que, para o cliente no final, não era o resultado

desejado, e isto, não ocorria somente para software de sistemas, mas para diversos modelos de negócios, segue a relação das principais áreas em que a Engenharia de Software é aplicada:

- **Software de sistema:** estes são, conjuntos de programas desenvolvidos para atender a outros programas ou necessidades fins. A exemplo temos: compiladores, editores e utilitários para gerenciamento de arquivos que processam estruturas de informações complexas, Outras aplicações de sistema, por exemplo, componentes de sistema operacional, drivers, softwares de redes de computadores, processadores de telecomunicações que processam dados amplamente indeterminados. Em ambos os casos, a área de software de sistemas é caracterizada por uma enorme interação com o hardware dos computadores, o uso intenso por múltiplos usuários, as operações concorrentes que requerem escalas de ordens altíssimas, compartilhamento de recursos e gestão de processo sofisticados; estruturas de dados complexas e múltiplas interfaces externas.
- **Software de aplicação:** são programas específicos que solucionam uma necessidade específica de negócio. Aplicados nesta área, processam dados comerciais ou técnicos de uma forma que facilite as operações comerciais ou tomadas de decisões administrativas/técnicas. Além das aplicações convencionais de processamento de dados, o software de aplicação é usado para controlar funções de negócio em tempo real, a exemplo, processamento de transações em pontos de venda, controle de processos de fabricação em tempo assíncrono.
- **Software científico / engenharia:** de uso e característico por algoritmos **number crunching**, para o processamento numérico pesado. As aplicações neste modelo vão da astronomia à vulcanologia, da análise de tensões na indústria automotiva à dinâmica orbital de ônibus espaciais, também da biologia molecular à fabricação automatizada. Entretanto, aplicações modernas dentro da área de engenharia/científica estão se afastando dos algoritmos numéricos convencionais. Projeto com o auxílio de computador, simulação de sistemas e outras aplicações interativas começaram a ter características de sistemas em tempo real e até mesmo de software de sistemas.
- **Software embutido:** residente em produtos específicos ou sistema é utilizado para implementar e controlar características e funções para o usuário final e para o próprio sistema. Executa funções limitadas e específicas, a exemplo: controle do painel de um forno micro-ondas ou fornece funções significativas e capacidade de controle, funções digitais de automóveis, tal como controle do nível de combustível, painéis de controle e sistemas de freios.
- **Software para linha de produtos:** projetado para prover a capacidade específica de utilização por muitos clientes diferentes. Pode focalizar um mercado limitado e particularizado: produtos para controle de estoques ou direcionar-se para os mercados de consumo de massa, processamento de texto, planilhas eletrônicas, computação gráfica, multimídia, entretenimento, gerenciamento de bancos de dados e aplicações financeiras pessoais e comerciais.
- **Aplicações para a Web:** chamadas de WebApps, essa categoria de software centralizada em redes, alcança uma vasta gama de aplicações. Em sua forma mais simples, as WebApps podem ser pouco

mais que um conjunto de arquivos de hipertexto interconectados, apresentando informações por meio de texto e gráficos limitados. Entretanto, com o aparecimento da Web 2.0, elas têm evoluído para sofisticados ambientes computacionais que não apenas fornecem recursos especializados, mas funções computacionais e conteúdo para o usuário final, como a integração de bancos de dados corporativos e aplicações comerciais.

- **Software de inteligência artificial:** faz uso de algoritmos não numéricos para solucionar problemas complexos que não são passíveis de computação ou de análise direta. Aplicações nessa área incluem: robótica, sistemas especialistas, reconhecimento de padrões de imagem e voz, redes neurais artificiais, prova de teoremas e jogos. Milhões de engenheiros de software em todo o mundo trabalham arduamente em projetos de software nas mais diversas categorias. Em alguns casos, novos sistemas estão sendo construídos, porém em outros casos as aplicações já existentes e estão sendo corrigidas, adaptadas e aperfeiçoadas. Não é incomum para um jovem engenheiro de software trabalhar em um programa mais velho que ele.
- **Computação mundial aberta:** o acelerado crescimento das redes sem fio pode, em breve, conduzir a uma verdadeira computação distribuída e ampliada, compartilhada e incorporada nos ambientes domésticos e comerciais. O desafio para os engenheiros de software será o de desenvolver sistemas e software aplicativo que permitam que dispositivos móveis, computadores pessoais e sistemas corporativos se comuniquem através de extensas redes. O que hoje já acontece em grande escala!
- **Netsourcing (recursos via Internet):** a Internet rapidamente tornou-se um grande e avassalador mecanismo computacional, como um provedor de conteúdo. O desafio para os engenheiros de software, consiste em arquitetar aplicações simples, isto é, planejamento financeiro, pessoal e sofisticados que forneçam benefícios aos mercados mundiais de usuários finais visados.
- **Software aberto:** tendência crescente que resulta na distribuição de código-fonte para aplicações de sistemas, por exemplo: sistemas operacionais, bancos de dados e ambientes de desenvolvimento, de forma que muitas pessoas possam contribuir para seu crescimento. O desafio para os engenheiros de software, consiste em construir um código fonte auto descritivo, mais importante ainda é, desenvolver técnicas que permitam que tanto os clientes quanto os desenvolvedores saibam quais alterações foram feitas e como se manifestam dentro do software.

Disciplinas da Engenharia de Software

Dentro do processo da Engenharia de Software, iremos destacar algumas disciplinas, lembrando que, não detalharemos esta parte, pois possuímos livros eletrônicos que descrevem cada uma delas em detalhes:

Requisitos de software: Na sistematização e engenharia de software, **análise de requisitos** engloba todas as tarefas que lidam com a investigação, definição e escopo de novos sistemas ou alterações. Análise de requisitos é parte importante do processo de desenvolvimento de softwares, na qual o engenheiro de requisitos e o analista de negócio, juntamente com engenheiros de sistema ou desenvolvedores de software, identificam as necessidades ou requisitos de um cliente. Uma vez que os requisitos do sistema tenham sido identificados, os projetistas de sistemas estarão preparados para projetar a solução.

A análise de requisitos é uma das primeiras atividades no desenvolvimento de software. O produto do seu trabalho é a especificação dos requisitos, esta, define o escopo do software em duas dimensões: **Requisito funcional e Requisito não-funcional**. É nesta fase que o analista faz as primeiras reuniões com os clientes e/ou usuários do software para conhecer as funcionalidades do sistema que será desenvolvido. Também, ocorre a maior parte dos erros, pois a falta de experiência dos clientes ou usuários faz com que nem sempre tenham claro em sua mente quais funcionalidades o software terá.

Outro conceito é: *Análise de Requisitos é o processo que envolve o estudo das necessidades do usuário para encontrar-se uma definição correta ou completa do sistema ou requisito de software.*

Projeto de software: parte responsável da engenharia de software que se encarrega de fazer todo o planejamento anterior ao desenvolvimento, incluindo a definição da arquitetura do software, e transformar tudo em um documento ou conjunto de documentos capazes de ser interpretados diretamente pelo programador. Para atingir este objetivo, o projetista deve mapear as estruturas e funcionalidades identificadas na análise de requerimentos dentro do contexto e das restrições da arquitetura, de forma a tornar possível a construção do software. Ao longo do tempo e nos diversos processos de software existentes, várias ferramentas foram idealizadas para facilitar e atingir este objetivo.

Construção de software: Um processo de desenvolvimento de software é um conjunto de atividades, parcialmente ordenadas, com a finalidade de obter um produto de software. Estudado dentro da área de Engenharia de Software, considerado um dos principais mecanismos para se obter software de qualidade e cumprir corretamente os contratos de desenvolvimento, sendo uma das respostas técnicas adequadas para resolver a crise do software. Este é um dos pilares da entrega de produto em tempo hábil.

Teste de software: O teste do software é a investigação do funcionamento do mesmo, a fim de fornecer informações sobre sua qualidade em relação ao contexto em que ele deve operar, se relaciona com o conceito de verificação e validação. Estas que foram levantadas através de requisitos funcionais e não funcionais. Isto inclui o processo de utilizar o produto para encontrar seus defeitos. O teste é um processo realizado pelo testador de software, que permeia outros processos da engenharia, e que envolve ações que vão do levantamento de requisitos até a execução do teste propriamente dito.

Manutenção de software: Em engenharia de software, manutenção de software é o processo de melhoria e otimização de um software já desenvolvido, lembrando que, um produto mal desenvolvido com certeza necessitará de manutenções constantes e reparos de defeitos. A manutenção do software é uma das fases do processo de desenvolvimento de software, ocorre após a entrada do software em produção. A manutenção pode ser preventiva ou mesmo corretiva. Também ocorre quando aponta para falhas no Processo de Desenvolvimento de Software - PDS que devem ser identificadas e tratadas, visando a melhoria contínua do processo. As anomalias são inseridas pelo processo em si ou pelo não entendimento dos requisitos do produto.

Gerência de configuração de software: A Gerência de configuração de software ou gestão de configuração de software é, a área da engenharia de software responsável por fornecer o apoio para o desenvolvimento deste. Suas atribuições básicas são o controle de versão, o controle de mudança e a auditoria das configurações.

Gerência de engenharia de software: A Engenharia de software é, a área da computação voltada à especificação, desenvolvimento, manutenção e criação de software, com a aplicação de tecnologias e práticas de gerência de projetos e outras disciplinas, visando organização, produtividade e qualidade. Atualmente, essas tecnologias e práticas englobam linguagens de programação, banco de dados, ferramentas, plataformas, bibliotecas, padrões de projeto de software, processo de software e qualidade de software. Além disso, a

engenharia de software deve oferecer mecanismos para se planejar e gerenciar o processo de desenvolvimento de um sistema computacional de qualidade e que atenda às necessidades de um requisitante de software.

Processos de Engenharia de Software: Um processo de Engenharia de Software é composto por um conjunto de passos de processo parcialmente ordenados, relacionados com artefatos, pessoas, recursos, estruturas organizacionais e restrições, tendo como objetivo produzir e manter os produtos de software finais requeridos.

Ferramentas e Métodos de Engenharia de Software: Outro ponto crucial é o uso de ferramentas CASE do inglês Computer-Aided Software Engineering. Essa classificação abrange toda ferramenta baseada em computadores que auxiliam as atividades da engenharia de software, desde a análise de requisitos, modelagem e até fase de testes. Os ambientes de desenvolvimento integrado (IDEs) têm maior destaque e suportam, entre outras:

- F) Editor.
- G) Debug.
- H) Geração de código.
- I) Modelagem.
- J) Deploy.
- K) Testes não automatizados.
- L) Testes automatizados.
- M) Refatoração (Refactoring).
- N) Gestão de Riscos nos projetos de Software.
- O) Uso da Prototipagem na Eng. de Requisitos.**

Qualidade de software: O significado de qualidade possui várias definições na literatura. O glossário do IEEE define qualidade como Grau de conformidade de um sistema, componente ou processo com os respectivos requisitos ou alternativamente, como Grau de conformidade de um sistema, componente ou processo com as necessidades e expectativas de clientes ou usuários.

Ambas as definições refletem os aspectos importantes da qualidade, diversos autores apresentam outras definições que, geralmente giram em torno dos temas de conformidade com os requisitos e atendimento das expectativas. Naturalmente, pode haver diferenças entre as aplicações dessas definições se os requisitos explícitos não refletirem corretamente as necessidades reais.