LISTA DE QUESTÕES

ENGENHARIA DE SOFTWARE

BANCA FCC

CONCURSO TRT 2014

Professor: Lúcio Camilo
Email: luciocamilo@gmail.com

QUESTÕES RUP

Questão 04 - FCC - 2012 - TRT - 6ª Região (PE) - Analista Judiciário - Tecnologia

A perspectiva prática sobre o RUP descreve as boas práticas da engenharia de software que são recomendadas para uso no desenvolvimento de sistemas. Dentre as práticas fundamentais recomendadas incluem-se

a) utilizar a arquitetura em cascata e efetuar programação em pares.

b) definir a funcionalidade do protótipo e avaliar o protótipo.

c) definir o esboço dos requisitos e estabelecer objetivos do protótipo.

d) utilizar arquiteturas baseadas em componentes e modelar os softwares visualmente.

e) desenvolver teste inicial a partir de cenários e utilizar frameworks de testes automatizados.

Questão 05 - FCC - 2009 - TJ-PA

O RUP é geralmente descrito por meio

a) da perspectiva dinâmica, apenas.

b) da perspectiva estática, apenas.

c) das perspectivas dinâmica e estática, apenas.

d) das perspectivas dinâmica e prática, apenas.

e) das perspectivas dinâmica, estática e prática.

Questão 06 - FCC - 2013 - DPE-SP - Agente de Defensoria - Analista de Sistemas

O modelo estabelecido para o RUP (*Rational Unified Process*) é composto por quatro fases, denominadas:

a) Requisitos, Implantação, Testes e Ambiente.

b) Análise, Projeto, Negócios e Comissionamento.

c) Concepção, Elaboração, Construção e Transição.

d) Partição, Integração, Testes e Operação.

e) Planejamento, Codificação, Integração e Configuração.

Questão 07 - FCC - 2011 - TCE-PR - Analista de Controle - Informática

A conclusão da análise, do design, do desenvolvimento e do teste de todas as funcionalidades necessárias ao sistema, no processo RUP, é um dos objetivos da fase de

a) iniciação.
b) elaboração.
c) integração.
d) construção.
e) transição.

Questão 08 - FCC - 2012 - MPE-PE - Analista Ministerial - Informática

A visão estática do RUP prioriza as atividades que ocorrem durante o processo de desenvolvimento. Na descrição do RUP, essas são chamadas de *workflows*. Existem seis *workflows* centrais, identificadas no processo e três de apoio, dentre os quais é possível citar os *workflows* de

a) Meio ambiente e Gerenciamento de projeto.
b) Concepção e Construção.
c) Transição e Iteração.
d) Plano de desenvolvimento e Conceito de operação.
e) Análise de Riscos e Operação e manutenção.

Questão 09 - FCC - 2011 - TRT - 23ª REGIÃO (MT) - Analista Judiciário - Tecnologia

A disciplina Gerenciamento de Projeto do RUP tem por finalidade fornecer um *framework* para gerenciamento de

I. Projetos específicos de *software.*

II. Riscos.

III. Orçamento.

IV. Contratos.

Está correto o que consta em

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

Uma disciplina do RUP que tem como uma de suas finalidades “assegurar que os clientes, usuários e desenvolvedores tenham um entendimento comum da organização-alvo”, a qual se relaciona com a disciplina Ambiente. Trata-se de

a) Requisitos.
b) Análise e *Design.*
c) Modelagem de Negócios.
d) Gerenciamento de Configuração e Mudança.
e) Gerenciamento de Projetos.

Questão 11 - FCC - 2010 - MPE-RN - Analista de Tec. da Informação – Eng. de Software

Considere:

I. Dirigido por caso de uso.
II. Orientado por quatro workflows.
III. Centrado em arquitetura.
IV. Distribuído em cinco fases.
V. Iterativo e incremental.

São características do Processo Unificado (UP) o que consta APENAS em

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

Questão 12 - FCC - 2007 - TRE-SE - Analista Judiciário - Análise de Sistemas

Considere as afirmativas abaixo.

I. O RUP é um processo iterativo.
II. Sob orientação do RUP, o desenvolvimento é centrado na arquitetura.
III. Sob a orientação do RUP, as atividades de desenvolvimento são orientadas por casos de uso.

É correto o que se afirma em

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

Questão 13 - FCC - 2010 – TRT-8

No Processo Unificado, uma descrição da arquitetura do *software, um documento de visão e um modelo de projeto são aplicáveis, respectivamente, nas fases:*

a) elaboração, concepção e construção.

b) concepção, concepção e elaboração.

c) construção, transição e concepção.

d) transição, construção e construção.

e) concepção, elaboração e transição.

Questão 14 - FCC - 2010 – MPE-SE

Pertencem à dimensão temporal do modelo iterativo RUP:

a) *Inception e Transition.*

b) *Implementation e Deployment.*

c) *Requirement e Configuration.*

d) *Elaboration e Implementation.*

e) *Elaboration e Test.*

Questão 15 - FCC - 2011 - TRE-AP

É um dos core “*supporting*” *workflows*, o

a) *Test.*
b) *Inception.*
c) *Analysis & Design.*
d) *Business modeling.*
e) *Configuration and Change Management.*

Questão 16 - FCC - 2011 - TRE-RN

O RUP produz artefatos

a) na fase de Transição, apenas.
b) em todas as suas fases.
c) na fase de Concepção, apenas.
d) na fase de Elaboração, apenas.
e) na fase de Construção, apenas.

Questão 17 - FCC - 2010 - MPE-RN

São respectivamente disciplina (Core Process Workflow) e fase (Phase) do RUP:

a) Concepção e Implantação.

b) Implementação e Elaboração.

c) Implantação e Requisitos.

d) Requisitos e Modelagem de Negócios.

e) Implementação e Teste.

Questão 18 - FCC - 2008 - TRF

O RUP possibilita o desenvolvimento

a) incremental e interativo, guiado por casos de uso e centrado na arquitetura do sistema.

b) interativo e centrado nos dados e informações do sistema.

c) incremental e interativo, em quatro camadas e centrado na estrutura dos dados do sistema.

d) interativo, guiado por casos de uso e centrado na infra-estrutura do sistema.

e) incremental e centrado na funcionalidade do sistema.

Questão 19 - FCC - 2009 - TJ-PA

Considere os artefatos de software abaixo.

I. Protótipo arquitetural executável.

II. Descrição da arquitetura.

III. Produto de software integrado na adequada plataforma.

A correta e respectiva associação desses artefatos com as fases do RUP é

a) Elaboração, Elaboração e Construção.

b) Elaboração, Concepção e Construção.

c) Concepção, Elaboração e Construção.

d) Concepção, Concepção e Elaboração.

e) Elaboração, Construção e Transição.

Questão 20 - FCC - 2009 - TJ-SE

NÃO é um dos *Core Process Workflows* do RUP o

a) *Implementation.*
b) *Environment.*
c) *Test.*
d) *Requirements.*
e) *Deployment.*

Uma estratégia de teste que é preferida por grande parte das equipes de software assume uma visão incremental do teste, começando com o teste das unidades individuais do programa, passando para os testes destinados a facilitar a integração de unidades e culminando com testes que usam o sistema concluído. No Processo Unificado (PU), os testes de unidades e testes de integração são realizados na fase de

(A) validação.

(B) elaboração.

(C) produção.

(D) transição.

(E) construção.

No Processo Unificado, a maior porção do core workflow denominado Analysis é executada na fase

(A) Elaboration.

(B) Construction.

(C) Implementation.

(D) Inception.

(E) Transition.

23 - FCC - 2012 – BANESE

De acordo com a arquitetura geral do RUP, a menor porção da disciplina de modelagem do negócio está relacionada com a fase

(A) Construction.
(B) Implementation.
(C) Inception.
(D) Elaboration.
(E) Transition.

24 - FCC - 2012 – Agencia Reguladora/CE

No RUP, uma das metas do workflow de requisitos é

(A) garantir que os clientes, usuários finais e desenvolvedores tenham um entendimento comum da organização.

(B) definir a organização do código em termos de implementação de subsistemas organizados em camadas.

(C) prover uma base para a estimativa de custos e tempo necessário para desenvolver um sistema.

(D) entender a estrutura e dinâmica da organização e derivar os requisitos de sistema necessários para suportar a organização.

(E) integrar em um sistema executável os resultados produzidos por times ou indivíduos.

Gabarito - RUP

	8 – A	15 – E	22 – A
	9 – A	16 – B	23 – C
	10 – C	17 – B	24 – C
4 – D	11 – C	18 – A	
5 – E	12 – A	19 – A	
6 – C	13 – A	20 – B	
7 – D	14 – A	21 – E	

QUESTÕES – MET. AGÉIS

Questão 04 - FCC- 2011 – TRT23/MT

NÃO se aplica à disciplina de desenvolvimento de *software extreme programming (XP):*

a) Usa notações próprias para construir os diversos produtos de trabalho do projeto.
b) Encoraja a refabricação para modificar um *sofware sem alterar o comportamento externo do código.*
c) Recomenda que dois programadores trabalhem juntos no mesmo computador para escrever um código.
d) Baseada em valores de simplicidade, comunicação, *feedback e coragem.*
e) Adota como um elemento-chave a criação de testes unitários antes da codificação começar.

Questão 05 - FCC - 2011 - TCE-PR - Analista de Controle - Informática

Dentre os papéis da metodologia ágil Scrum está o *Scrum Master*. NÃO se inclui entre as funções deste papel

a) remover impedimentos para o progresso do time de desenvolvimento.
b) comunicar claramente a visão, metas e itens de backlog do produto ao time de desenvolvimento.
c) determinar para o time de desenvolvimento como os itens de *backlog* devem ser convertidos em potenciais funcionalidades para entrega.
d) entender o planejamento de produto de longo termo em um ambiente empírico.
e) ajudar os empregados e envolvidos com o projeto no entendimento e promulgação de *Scrum* e produtos empíricos.

Questão 06 - FCC - 2012 - MPE-AP - Analista Ministerial - Tecnologia da Informação

O *Extreme Programming (XP*) é, talvez, o mais conhecido e mais utilizado dos métodos ágeis. Dentre suas práticas se encontram programação em pares, integração contínua, refatoração e

a) propriedade coletiva, que garante uma participação nos lucros aos membros da equipe de desenvolvimento, técnica que incentiva e aumenta o desempenho de toda a equipe.

b) envolvimento do cliente apenas na fase final do sistema, fator que difere de outras metodologias como SCRUM e TDD e confere agilidade ao processo de desenvolvimento.

c) processo de desenvolvimento contínuo, em que a equipe se mantém focada no sistema até que uma funcionalidade específica seja entregue, comumente agregando horas extras ao turno de trabalho.

d) utilização de técnicas de ofuscação do código fonte, trazendo segurança e garantindo que apenas a equipe de desenvolvimento poderá ter acesso a este código

e) desenvolvimento incremental e sustentado por meio de pequenos e frequentes releases do sistema. Os requisitos são baseados em cenários ou em simples histórias de clientes.

Questão 07 - FCC - 2011 - TCE-PR - Analista de Controle - Informática

Na metologia *Scrum, Sprint* é uma iteração de duração menor ou igual a um mês, onde uma parte incremental e funcional do produto está potencialmente pronta para entrega. É INCORRETO afirmar que, nessa fase,

a) o escopo pode ser esclarecido e renegociado entre o time de desenvolvimento e o proprietário do produto.
b) nenhuma alteração que afetaria a meta do *Sprint* é efetuada.
c) a composição do time de desenvolvimento permanece constante.
d) as metas de qualidade não diminuem.
e) o *Sprint* pode ser cancelado por decisão do *Scrum Master.*

Questão 08 - FCC - 2011 - INFRAERO – Analista Sistemas - Arquitetura de Software

Um dos principais conceitos do Scrum para atacar a complexidade do desenvolvimento e gerenciamento de software é a implantação de um controle descentralizado, capaz de lidar mais eficientemente com contextos pouco previsíveis. Para tanto, o gerenciamento é distribuído por meio de três agentes independentes que são:

a) *Product Owner, Scrum Team* e *Scrum Master*.
b) *Product Owner, Product Backlog* e *Planning Meeting*.
c) *Product Owner, Sprint* e *Planning Meeting*.
d) *Sprint, Scrum Master* e *Planning Meeting*.
e) *Sprint, Scrum Team* e *Product Backlog.*

Questão 09 - FCC - 2011 - TRT - 23ª REGIÃO (MT) - Técnico Judiciário - Tecnologia

No desenvolvimento de *software* em *Extreme Programming* (XP) há uma confiança muito grande na sinergia entre as práticas, já que os pontos fracos de cada uma são superados pelos pontos fortes de outras. Dentre elas, aquela em que o código fonte não tem dono e ninguém precisa solicitar permissão para poder modificá-lo, permitindo, assim, que a equipe conheça todas as partes do sistema, é chamada de

a) *Whole Team* (Time Coeso).

b) *Sustainable Pace* (Ritmo Sustentável).

c) *Pair Programming* (Programação em Pares).

d) *Collective Ownership* (Posse Coletiva).

e) *Coding Standards* (Padrões de Codificação).

Questão 10 - FCC- 2012 – TRF2

Segundo Roger S. Pressman, em seu livro Engenharia de *Software, 7a edição, os princípios do Scrum são consistentes com o manifesto ágil e são usados para orientar as atividades de desenvolvimento dentro de um processo que incorpora as atividades estruturais de requisitos, análise, projeto, evolução e entrega. Em cada atividade metodológica, ocorrem tarefas a realizar dentro de um padrão de processo chamado*

(A) *process backlog.*

(B) *scrum master.*

(C) *product owner.*

(D) *backlog.*

(E) *sprint.*

Questão 11 - FCC- 2012 – TER/CE

No SCRUM, *sprint é*

(A) um representante dos *stakeholders e do negócio.*

(B) uma lista de requisitos que tipicamente vêm do cliente.

(C) uma lista de itens priorizados a serem desenvolvidos para um *software.*

(D) uma iteração que segue um ciclo (PDCA) e entrega incremento de *software pronto.*

(E) um conjunto de requisitos, priorizado pelo *Product Owner.*

Questão 12 - FCC- 2012 – MPE/AP

Na metodologia de desenvolvimento SCRUM, o proprietário do produto *(Product Owner) é responsável por maximizar o valor do produto e o trabalho da equipe de desenvolvimento.*

O proprietário do produto é a única pessoa responsável pela manutenção do *Backlog do produto. Este gerenciamento inclui*

a) assegurar que a equipe de desenvolvimento compreenda os itens do *Backlog do produto no nível necessário.*

b) encontrar técnicas para a manutenção efetiva do *Backlog do produto e transmitir essas técnicas para a equipe de desenvolvimento.*

c) comunicar, nas reuniões diárias, as metas e itens do *Backlog do produto para a equipe de desenvolvimento.*

d) treinar o time Scrum para que crie, de forma clara e precisa, os itens do *Backlog do produto.*

e) entender o planejamento do produto a longo termo e de forma empírica.

Questão 13 - FCC - 2011 - TRE-RN - Analista Judiciário - Análise de Sistemas

Considere as seguintes características:

I. Propriedade coletiva.

II. Integração contínua.

III. Metáfora.

Dentre as práticas componentes da Extreme Programming, aplica-se o que consta em

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

Questão 14 - FCC - 2010 - TRE-RS - Técnico Judiciário - Programação de Sistemas

No contexto das regras do SCRUM, é correto afirmar:

a) Durante a realização do *Sprint*, o *Backlog* pode ser modificado por qualquer um dos elementos da equipe, desde que acordado nas reuniões semanais.

b) O *Sprint* deve ser realizado num período não superior a 30 dias e ter um objetivo bem claro, baseado no *Backlog.*

c) Modificação no *Backlog* é prerrogativa do *Scrum Master*, quando achar necessário, em qualquer momento no decorrer do *Sprint.*

d) Não é possível dissolver um *Sprint*. Se houver algum risco de ele tomar um rumo não desejável, novas funcionalidades devem ser implementadas para garantir o prazo do projeto.

e) O foco na produtividade se estende às *Scrum meetings* e a conversação é pautada em discussões por toda a equipe.

O conceito de *sprint* aplica-se ao modelo ágil do processo de engenharia de *software* denominado

a) XP.
b) DAS.
c) DSDM.
d) Scrum.
e) Crystal.

Questão 16 - FCC - 2010 - TRF - 4ª REGIÃO - Analista Judiciário - Tecnologia

A Extreme Programming (XP) baseia-se em 12 práticas, que são um conjunto de atividades que deverão ser seguidas pelas equipes que desejam utilizar a XP. Na prática do Jogo do Planejamento, as funcionalidades são descritas em pequenos cartões que são conhecidos como

a) cartões de requisitos.

b) cartões de planejamento.

c) cartões chave.

d) cartões inteligentes.

e) histórias de usuário.

Questão 17 - FCC - 2009 - TRT - 15ª Região - Analista Judiciário - Tecnologia

Histórias de usuários na atividade de planejamento, encorajamento de uso de cartões CRC e de refabricação, reuniões em pé e programação em pares são características típicas do modelo de processo de software

a) XP.

b) SCRUM.

c) DSDM.

d) DAS.

e) MVC.

Questão 18 – FCC - 2013 – TRT12

SCRUM é um *framework* baseado no modelo ágil. No SCRUM,

a) o *scrum team* é a equipe de desenvolvimento, necessariamente dividida em papéis como analista, *designer* e programador. Em geral o *scrum team* tem de 10 a 20 pessoas.

b) as funcionalidades a serem implementadas em cada projeto (requisitos ou histórias de usuários) são mantidas em uma lista chamada de *scrum board*.

c) o *scrum master* é um gerente no sentido dos modelos prescritivos. É um líder, um facilitador e um solucionador de conflitos. É ele quem decide quais requisitos são mais importantes.

d) um dos conceitos mais importantes é o *sprint* , que consiste em um ciclo de desenvolvimento que, em geral, tem duração de 4 a 7 dias.

e) o product *owner* tem, entre outras atribuições, a de indicar quais são os requisitos mais importantes a serem tratados em cada *sprint* . É responsável por conhecer e avaliar as necessidades dos clientes.

19 - FCC - 2013 – Defensoria RS

Sobre os processos ágeis de desenvolvimento de software XP e Scrum, considere:

I. Emprega uma abordagem orientada a objetos como seu paradigma de desenvolvimento preferido e envolve um conjunto de regras e práticas constantes no contexto de quatro atividades metodológicas: planejamento, projeto, codificação e testes.

II. Seus princípios são usados para orientar as atividades de desenvolvimento dentro de um processo que incorpora as se- guintes atividades estruturais: requisitos, análise, projeto, evolução e entrega. Em cada atividade metodológica ocorrem tarefas a realizar dentro de um padrão de processo chamado sprint.

III. Faz uso do teste de unidades como sua tática de testes primária. À medida que cada classe é desenvolvida, a equipe de- senvolve um teste de unidade para exercitar cada operação de acordo com a sua funcionalidade especificada. À medida que um incremento é entregue a um cliente as histórias de usuários ou casos de uso implementados pelo incremento são usados como base para testes de aceitação.

IV. O jogo do planejamento se inicia com a atividade de ouvir (que constitui uma atividade de levantamento de requisitos). Essa atividade conduz à criação de um conjunto de histórias de usuários que descreve o resultado, as características e a funcionalidade requisitados para o software a ser construído.

A associação correta entre cada item e o respectivo processo ágil é

	XP	Scrum
A	I e III	II e IV
B	II e IV	I e III
C	III e IV	I e II
D	II, III e IV	I
E	I, III e IV	II

20 - FCC - 2014 – SABESP

A primeira grande divisão de um processo é a fase. Uma fase é um período de tempo no qual determinadas atividades com objetivos bem específicos são realizados. Sobre as fases dos principais modelos de processos, analise:

I. Alguns processos, como o Modelo Espiral e suas variantes, têm fases sequenciais, ou seja, com o passar do tempo o pro- cesso de desenvolvimento passa de uma fase a outra, como requisitos, análise, programação, testes e implantação.

II. Alguns modelos de processo, como o Modelo Cascata, Modelo de Prototipação Evolucionária e Modelos Ágeis têm fases cíclicas, ou seja, o desenvolvimento passa repetidamente de uma fase para outra, formando um ciclo repetitivo de fases até a finalização do projeto.

III. O Processo Unificado (UP) é estruturado em quatro fases (embora algumas variantes tenham até seis fases), que são se- quenciais no tempo. Dentro de cada fase, as atividades são organizadas de forma cíclica, ou seja, existem ciclos itera- tivos dentro das fases, mas elas são sequenciais.

Está correto o que se afirma APENAS em

(A) II.

(B) II e III.

(C) III.

(D) I e III.

(E) I e II.

21 - FCC - 2014 – TRF

Scrum é um modelo utilizado no desenvolvimento ágil de software. No Scrum um dos conceitos mais importantes é o sprint, que consiste em um ciclo de desenvolvimento que, em geral, vai de duas semanas a um mês. No início de cada sprint é feito um ..... I , no qual a equipe prioriza os elementos do ..... II a serem implementados e transfere esses elementos para o ..... III , ou seja, a lista de funcionalidades a serem implementadas no ciclo que se inicia. A equipe se compromete a desenvolver as funcionalidades e o ..... IV se compromete a não trazer novas funcionalidades durante o mesmo sprint.

As lacunas I, II, III e IV são preenchidas, correta e respectivamente, por

(A) sprint burndown, product backlog, sprint backlog, scrum team.

(B) sprint planning meeting, product backlog, sprint backlog, product owner.

(C) scrum planning, sprint backlog, product backlog, product owner.

(D) sprint planning meeting, product backlog, sprint backlog, scrum master.

(E) scrum daily meeting, product backlog, sprint backlog, scrum master.

SCRUM, o processo de desenvolvimento inicia com uma reunião de planejamento na qual o Product Owner e a equipe decidem, em conjunto, o que deverá ser imple- mentado do Product Backlog. Assim, a equipe planeja seu trabalho, definindo o Sprint Backlog, na

(A) primeira parte da Sprint Planning Meeting.

(B) segunda parte da Sprint Planning Meeting.

(C) terceira parte da Sprint Planning Meeting.

(D) Sprint.

(E) Sprint Burndown.

23 - FCC - 2014 – TRT 2ªRegião

Há diversos processos e práticas ágeis de desenvolvimento de software. Considere:

I. Seu objetivo é criar um “código limpo que funcione”. Trabalha com a estratégia Red − Green − Refactor:

- − Codifique o teste;
- − Faça-o compilar e executar. O teste não deve passar (Red).
- − Implemente o requisito e faça o teste passar (Green).
- − Refatore o código (Refactor).

II. Suas práticas, regras e valores garantem um agradável ambiente de desenvolvimento de software para os seus seguidores, que são conduzidos pelos princípios básicos:

- − Comunicação − manter o melhor relacionamento possível entre clientes e desenvolvedores, preferindo conversas pessoais a outros meios de comunicação;
- − Simplicidade − implementar apenas requisitos atuais, evitando adicionar funcionalidades que podem ser importantes somente no futuro;
- − Feedback − o desenvolvedor terá informações constantes do cliente e do código, em que testes constantes indicam os erros tanto individuais quanto do software integrado;
- − Coragem − encorajar as pessoas que não possuem facilidade de comunicação e bom relacionamento interpessoal, encorajar a equipe a experimentar e buscar novas soluções, além de encorajar a obtenção de feedback do cliente.

III. Objetiva capturar os critérios de aceitação para as funcionalidades em desenvolvimento. Trabalha com as seguintes etapas:

- − Discutir (Discuss): discussão colaborativa com a equipe visando elicitar os critérios de aceitação.
- − Refinar (Distill): refinamento dos critérios de aceitação em um conjunto concreto de cenários/exemplos de uso descrevendo o comportamento esperado da aplicação em uma linguagem comum a todos os membros da equipe.
- − Desenvolver (Develop): transformação dos testes de aceitação (descrevendo o comportamento esperado do software) em testes/especificação automatizados.

IV. Suas práticas incluem:

- − Envolver as partes interessadas no processo através de Outside-in Development. − Usar exemplos para descrever o comportamento de uma aplicação ou unidades de código.
- − Automatizar os exemplos para prover um feedback rápido e testes de regressão.
- − Usar o verbo deve (should) ao descrever o comportamento de software para ajudar a esclarecer responsabilidades e permitir que funcionalidades sejam questionadas.
- − Usar dublês de teste (mocks, stubs, fakes, dummies, spies) para auxiliar na colaboração entre módulos e códigos que ainda não foram escritos.

Os processos ágeis I, II, III e IV são, correta e respectivamente, denominados:

(A) BDD − DDD − ATDD − XP
(B) TDD − BDD − DDD − XP
(C) ATDD − XP − DDD − BDD
(D) ATDD − BDD − TDD − DDD
(E) TDD − XP − ATDD − BDD

Ana foi contratada em uma empresa para efetuar trabalhos de desenvolvimento relacionados à área de informática. Logo no primeiro dia foi convidada a participar de uma reunião que é efetuada diariamente, de apenas 15 minutos. Todos os participantes ficam em pé e ela é conduzida pelos próprios desenvolvedores. Durante este pequena reunião, foram abordados o que cada desenvolvedor conseguiu concluir desde a última reunião, o que ele pretende efetuar até a próxima e, o que Ana achou muito importante, o que está impedindo que este desenvolvedor prossiga com seu trabalho. Ana foi informada que esta reunião pertence ao método ágil

(A) Jerkins, e que o nome dado a esta reunião é Sprint.

(B) Kanban, e as questões efetuadas são chamadas de artefatos.

(C) Scrum, e que o nome dado a esta reunião é Daily Scrum.
(D) Sprint, e que as questões efetuadas são chamadas de Backlog.

(E) XP, e que tanto a reunião quanto as perguntas são denominadas Interação Contínua.

25 - FCC - 2013 – TRT 9ªRegião

Os modelos de processos tradicionais surgiram em um cenário muito diferente do atual, baseado em mainframes e terminais remotos. Já os modelos de processos ágeis são adequados para situações atuais nas quais a mudança de requisitos é frequente. Dentre os modelos de processos ágeis mais comuns temos: Extreme Programming (XP), Scrum e Feature Driven Development (FDD).

Algumas das práticas e características desses modelos de processo são descritas a seguir:

I. Programação em pares, ou seja, a implementação do código é feita em dupla.

II. Desenvolvimento dividido em ciclos iterativos de até 30 dias chamados de sprints.

III. Faz uso do teste de unidades como sua tática de testes primária.

IV. A atividade de levantamento de requisitos conduz à criação de um conjunto de histórias de usuários.

V. O ciclo de vida é baseado em três fases: pre-game phase, game-phase, post-game phase.

VI. Tem como único artefato de projeto os cartões CRC.

VII. Realiza reuniões diárias de acompanhamento de aproximadamente 15 minutos.

VIII. Define seis marcos durante o projeto e a implementação de uma funcionalidade: walkthroughs do projeto, projeto, inspeção do projeto, codificação, inspeção de código e progressão para construção.

IX. Os requisitos são descritos em um documento chamado backlog e são ordenados por prioridade.

A relação correta entre o modelo de processo ágil e a prática/característica é:

	XP	Scrum	FDD
A	II, V e VII	II, IV e VIII	VII e IX
B	I, III, IV e VI	II, V, VII e IX	VIII
C	II, VII e VIII	III, IV, VI e IX	I e V
D	II, VII e VIII	I, III, IV e V	VI e IX
E	I, III, IV e VI	II, VIII, VII e IX	V

Não se trata de uma prática específica do XP (Extreme Programming)

(A) Test Driven Development.

(B) Refactoring.

(C) Sprint backlog.

(D) Pair Programming.

(E) Simple Design

Gabarito Questões – MET AGÉIS

	9 – D	17 – A	25 - B
	10 – E	18- E	26 - C
	11 - D	19 – E	
4 – A	12 - A	20 – C	
5 – C	13 – E	21 – B	
6 – E	14 – B	22 – B	
7 – E	15 – D	23 – E	
8 – A	16 - E	24 – C	

QUESTÕES - TESTES

Questão 01 - FCC - 2010 - TRT - 9ª REGIÃO (PR) - Analista Judiciário - Tecnologia da Informação

O teste de sistema que força o software a falhar de diversos modos e verifica o retorno do processamento dentro de um tempo pré-estabelecido é um tipo de teste de

a) Integração.

b) Estresse.

c) Recuperação.

d) Desempenho.

e) Segurança.

Questão 02 - FCC - 2009 - TRE-PI - Técnico Judiciário - Programação de Sistemas

Também conhecido por teste estrutural ou orientado à lógica, é uma técnica de teste de software que trabalha diretamente sobre o código fonte do componente de software para avaliar aspectos, tais como, teste de condição, teste de fluxo de dados, teste de ciclos e teste de caminhos lógicos. Trata-se da técnica de teste

a) da Caixa-branca.

b) da Caixa-cinza.

c) da Caixa-preta.

d) de Integração.

e) de Regressão.

Questão 03 - FCC - 2009 - MPE-SE - Analista do Ministério Público – Especialidade Análise de Sistemas

A execução de um sistema com o objetivo de encontrar falhas sob condições que demandam recursos em quantidade, frequência ou volume anormais é definida como

a) *payload*.

b) teste de estresse.

c) teste de desempenho.

d) latência da falha.

e) *workload*.

Questão 04 - FCC - 2009 - TRT - 15ª Região - Analista Judiciário - Tecnologia da Informação

Os testes de integração têm por objetivo verificar se

a) os módulos testados produzem os mesmos resultados que as unidades testadas individualmente.

b) os módulos testados suportam grandes volumes de dados.

c) as funcionalidades dos módulos testados atendem aos requisitos.

d) os valores limites entre as unidades testadas individualmente são aceitáveis.

e) o tempo de resposta dos módulos testados está adequado.

Questão 05 - FCC - 2008 - TRT - 18ª Região (GO) - Analista Judiciário - Tecnologia da Informação

Uma sistemática para construção da arquitetura do software enquanto, ao mesmo tempo, conduz ao descobrimento de erros associados às interfaces é a estratégia de teste de software denominada de

a) sistema.

b) unidade.

c) validação.

d) arquitetura.

e) integração.

Questão 06 - FCC - 2013 - DPE-SP - Agente de Defensoria - Analista de Sistemas

O teste de *software* constitui-se em uma etapa importante no ciclo de desenvolvimento de *software*. Uma das características mais importantes de um conjunto de testes de *software*, adequadamente planejados, é

a) provar a correção integral no programa sob teste.

b) ter alta probabilidade de detectar erros no programa sob teste.

c) ter grande redundância, a fim de testar mais de uma vez cada linha do programa sob teste.

d) ser de alta complexidade, pois assim pode-se cobrir todo o programa sob teste com apenas um teste.

e) ser ocultado da equipe de desenvolvimento do *software,* pois esta pode querer impedir sua aplicação.

Questão 07 - FCC - 2012 - TCE-AM - Analista de Controle Externo - Tecnologia da Informação

Sobre teste de software considere:
I. Uma estratégia de teste que é escolhida por grande parte das equipes de software adota uma visão incremental do teste, começando com o teste de unidades individuais de programa, avançando para testes projetados a fim de facilitar a integração das unidades e culmina com testes que exercitam o sistema construído.
II. O teste de unidade focaliza o esforço de verificação na menor unidade de projeto do *software* - o componente ou módulo de *software.* Usando a descrição de projeto no nível de componente como guia, caminhos de controle importantes são testados para descobrir erros dentro dos limites do módulo.
III. O teste de unidade é normalmente considerado um apêndice ao passo de codificação. O projeto de teste de unidade pode ser realizado antes que o código seja iniciado ou depois de o código-fonte ter sido gerado.
IV. O teste de integração é uma técnica sistemática para construir a arquitetura do *software* enquanto, ao mesmo tempo, conduz testes para descobrir erros associados às interfaces. O objetivo é, a partir de componentes testados no nível de unidade, construir uma estrutura de programa determinada pelo projeto.

Está correto o que se afirma em
a) I, II, III e IV.
b) I, II e IV, apenas.
c) II, III e IV, apenas.
d) III e IV, apenas.
e) I e III, apenas.

Questão 08 - FCC - 2012 - TRT - 6ª Região (PE) - Analista Judiciário - Tecnologia da Informação

Sobre testes de sistemas, considere:

I. Testes de cenário são úteis pois podem garantir que não restam erros no sistema. Neste ponto diferem dos testes de componentes que apenas garantem a integridade de módulos isolados do sistema, mas não garantem que a totalidade do sistema está isenta de erros.

II. Testes de desenvolvimento incluem testes unitários, nos quais são testados objetos e métodos específicos; testes de componentes, nos quais são testados diversos grupos de objetos; testes de sistema, nos quais são testados sistemas parciais e sistemas completos.

III. Os testes de usuário podem ser divididos em três fases: teste alfa, em que os usuários do *software* trabalham com a equipe de desenvolvimento para efetuar testes no local do desenvolvedor; teste beta, em que um release de *software* é disponibilizado aos usuários para que possam experimentar e levantar os problemas descobertos com os desenvolvedores do sistema; teste de sistema, em que os clientes testam um sistema para decidir se ele está pronto para ser implantado no ambiente de trabalho.

Está correto o que se afirma em

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

Questão 09 - FCC - 2012 - TJ-RJ - Analista Judiciário - Análise de Sistemas

No que se refere a testes de *software,* é correto afirmar que

a) o teste de operação é a fase onde é testada a ergonomia da interface de uso do *software*.

b) o teste da caixa preta (teste funcional), baseia-se em analisar os arquivos de *log* do sistema procurando por mensagens de funcionamento inconsistente.

c) um teste bem sucedido é um teste que não encontra nenhum erro no *software*.

d) o teste da caixa branca (teste estrutural), baseia-se em testar as estruturas do código fonte, como comandos condicionais e de repetição.

e) um caso de teste é uma categoria de possíveis resultados na execução de testes.

Questão 10 - FCC - 2012 - TRT - 11ª Região (AM) - Analista Judiciário - Tecnologia da Informação

Considere:
O objetivo é executar o sistema sob o ponto de vista de seu usuário final, varrendo as funcionalidades em busca de falhas em relação aos objetivos originais. Os testes são executados em condições similares àquelas que um usuário utilizará no seu dia-a- dia de manipulação do sistema.
A afirmativa refere-se ao teste de

a) aceitação.
b) sistema.
c) unidade.
d) operação.
e) integração.

Questão 11 - FCC - 2012 - TRT - 11ª Região (AM) - Técnico Judiciário - Tecnologia da Informação

NÃO É uma técnica típica de teste de caixa preta:

a) teste de tabela de decisão.

b) teste de todos os pares.

c) teste de integração.

d) teste de caso de uso.

e) tabelas de estado de transição.

Questão 12 - FCC - 2011 - TRE-PE - Analista Judiciário - Análise de Sistemas

Com relação aos testes de *software*, é correto afirmar:

a) Um princípio muitas vezes adotado ao testar um *software* é o de Pareto. Ele afirma que existe um forte desequilíbrio entre causas e efeitos, entre esforços e resultados e entre ações e objetivos alcançados.

b) Testes sempre podem mostrar a ausência de erros.

c) Para que o resultado de um teste de *software* seja confiável, é preciso garantir que os casos de teste utilizados cubram um número reduzido de possibilidades de execução.

d) Um *software* que produz saídas corretas deve ser aprovado, pois isso demonstra que todos os erros foram corrigidos.

e) Um programador deve testar seu próprio código porque facilmente conseguirá criar um caso de teste que rompe com a lógica de funcionamento do seu código.

Questão 13- FCC - 2011 - TRT - 14ª Região (RO e AC) - Técnico Judiciário - Tecnologia da Informação

Garantir o funcionamento correto do *software* para atender as expectativas do cliente é o objetivo da homologação de sistemas. Nessa fase, que precede à implantação, os testes mais comuns são os testes:

a) funcionais, de usabilidade e de aceitação.

b) de unidade, de iteração e de Integração.

c) de volume, de integridade e de aceitação.

d) da caixa-branca, de carga e de configuração.

e) de unidade, de carga e de integridade.

14 - FCC - 2013 – Defensoria RS

Considere:

Caso 1: Pedro foi contratado para realizar testes de software na empresa B. Realizava um conjunto de testes na interface do software focados em exercitar os requisitos funcionais. Na bateria de testes que realizava, procurava encontrar funções incorretas ou faltando, erros de interface, erros em estruturas de dados, erros em acesso a base de dados externas, erros de comportamento e de desempenho e erros de inicialização e término.

Caso 2: Paulo foi contratado para realizar testes de software na empresa C. Realizava testes nos caminhos lógicos do software e nas colaborações entre componentes exercitando conjuntos específicos de condições e/ou ciclos. Testava todos os caminhos independentes dos módulos pelo menos uma vez, exercitava as decisões lógicas nos seus estados verdadeiro ou falso e exercitava estruturas internas para assegurar a sua validade.

Pedro realizava testes

(A) caixa-branca e Paulo realizava testes caixa-preta.

(B) de caminho básico e Paulo realizava testes de condição.

(C) de unidade e Paulo realizava testes de integração.

(D) caixa-preta e Paulo realizava testes caixa-branca.

(E) de ciclo e Paulo realizava testes de fluxo de dados

15 - FCC - 2012 – BANESE

De acordo com Pressman, é uma técnica sistemática para construir a arquitetura do software enquanto, ao mesmo tempo, conduz testes para descobrir erros associados às interfaces. Trata-se, especificamente, de

(A) arquitetura top-down.

(B) teste de mesa.

(C) teste de integração.

(D) análise bottom-up.

(E) teste funcional

16 - FCC – 2011 – INFRAERO

Executa um sistema de forma a demandar recursos em volume ou frequência acima do normal. Trata-se de

A) validação de unidade de implementação.

B) teste beta.

C) teste de estresse.

D) validação de integração.

E) teste de desempenho.

17 - FCC – 2011 – INFRAERO

NÃO se trata de uma categoria de erros encontrados por meio de teste caixa-preta:

A) Conjunto básico de caminhos de execução.

B) Funções incorretas ou omitidas.

C) Acesso à base de dados externa.

D) Comportamento ou desempenho.

E) Iniciação e término.

18 - FCC – 2011 – INFRAERO

Considere a seguinte definição de uma característica de testabilidade (Pressman): Controlando o escopo do teste, podemos isolar problemas mais rapidamente e realizar retestagem mais racionalmente. O sistema de software é construído por meio de módulos independentes, que podem ser testados independentemente. Trata-se da característica:

A) estabilidade.

B) simplicidade.

C) operabilidade.

D) controlabilidade.

E) decomponibilidade.

19 - FCC – 2012 – PSP

Sobre testes de software e avaliação de qualidades de testes é INCORRETO afirmar que

A) teste de aceitação é um processo de teste de usuário no qual o objetivo é decidir se o software é bom o suficiente para ser implantado e usado em seu ambiente operacional.

B) testes de desenvolvimento são de responsabilidade da equipe de desenvolvimento de software, enquanto outra equipe deve ser responsável por testar o sistema antes que ele seja liberado para os clientes.

C) testes de desenvolvimento incluem testes unitários, nos quais são testados objetos e métodos específicos.

D) sempre que possível é recomendado escrever testes automatizados. Os testes são incorporados em um programa que pode ser executado cada vez que uma alteração é feita para um sistema.

E) testes podem anunciar a presença de defeitos em um programa e podem demonstrar que não existem defeitos remanescentes.

20 - FCC – 2013 – ALE/RN

O teste de software é destinado a mostrar que um programa faz o que é proposto a fazer e a descobrir seus defeitos antes do uso. O processo de teste tem dois objetivos distintos:

1) Demonstrar ao desenvolvedor e ao cliente que o software atende a seus requisitos.
2) Descobrir situações em que o software se comporta de maneira incorreta, indesejável ou de forma diferente das especificações.

Desse modo, é correto afirmar que

A) não é objetivo final dos processos de verificação validar os requisitos de especificação que não reflitam os desejos ou necessidades dos clientes.

B) os testes podem mostrar a presença de erros e sua ausência.

C) o objetivo de todo teste é verificar se ele atende apenas aos requisitos funcionais.

D) verificação e validação não são a mesma coisa em relação a testes de sistema.

E) os testes podem demonstrar que um determinado software está livre de defeitos

21 - FCC – 2013 – DPE/SP

Para aplicações convencionais, o software é testado a partir de duas perspectivas diferentes: a lógica interna do programa é exercitada usando técnicas de projeto de caso de teste I e os requisitos de software são exercitados usando técnicas de projeto de casos de teste II.

O teste I fundamenta-se em um exame rigoroso do detalhe procedimental. Os caminhos lógicos do software e as colaborações entre componentes são testados exercitando conjuntos específicos de condições e/ou ciclos.

O teste II faz referência a testes realizados na interface do software. Esse tipo de teste examina alguns aspectos fundamentais de um sistema, com pouca preocupação em relação à estrutura lógica interna do software.

As lacunas I e II são preenchidas correta e respectivamente, com:

A) de caminho básico - caixa-de-vidro

B) alfa - beta

C) caixa branca - caixa preta

D) de ciclo - de usabilidade

E) unitário - de interface

22 - FCC – 2014 – METRO/SP

Os testes de software podem ser feitos de forma manual ou automatizada. A automação de testes

A) não exige profissionais muito qualificados, pois as ferramentas de teste são projetadas com alto grau de usabilidade, e podem ser operadas por profissionais com pouca experiência.

B) funcionais só vale a pena se for repetida poucas vezes, pois o custo é muito alto. O ideal é automatizar os testes raramente executados.

C) não é indicada logo no início do processo de desenvolvimento do software, mas sim quando o software possuir certa estabilidade.

D) substitui completamente a execução dos testes manuais, uma vez que as ferramentas de teste trazem um conjunto de conhecimentos bem maior do que normalmente teria uma equipe inteira de testadores manuais.

E) se torna viável quando a totalidade dos testes é automatizada, aumentando a eficiência, reduzindo os custos e permitindo que o software seja colocado em produção na metade do tempo normal.

23 - FCC – 2013 – ALE/RN

Com relação aos tipos de testes de software, considere:

I. Testes baseados em requisitos são uma abordagem sistemática para projeto de casos de teste em que se considera cada requisito e deriva-se um conjunto de testes para eles. São mais uma validação do que um teste de defeitos.

II. Testes de release são feitos pela própria equipe de desenvolvimento e devem centrar-se na descoberta de bugs no sistema, nos quais os casos de teste são projetados para expor os defeitos.

III. Testes de desenvolvimento incluem testes unitários, nos quais se testa objetos e métodos específicos; testes de componentes, em que se testa diversos grupos de objetos; e testes de sistema, nos quais se testa sistemas parciais ou completos.

IV. Teste beta é um tipo de teste de usuário em que os usuários do software trabalham com a equipe de desenvolvimento para testar o software no local do desenvolvedor.

Está correto o que se afirma APENAS em

A) I e III.

B) II e IV.

C) I e II.

D) III e IV.

E) I, II e III.

24 - FCC – 2012 – MPE/RN

Sobre testes de unidade, considere:

I. O objetivo é utilizar uma pequena parte de código responsável por alguma funcionalidade muito específica dentro do software a ser desenvolvido, e testá-lo para garantir que ele se comporta exatamente como planejado sob várias condições.

II. Em testes convencionais, que podem ser feitos de forma manual ou automatizada, a validação de uma funcionalidade ocorre tipicamente depois que o software é desenvolvido, sendo que neste momento é quase impossível resolver problemas críticos ou de arquitetura de uma forma rápida. Com testes unitários o trabalho do programador é validado muito mais rapidamente por meio de testes de módulos pequenos do software, assim que eles são desenvolvidos, permitindo mudanças rápidas no código caso defeitos ou desvios de arquitetura sejam detectados.

III. Esse método permite que sejam testadas partes do software que geralmente não são expostas diretamente ao usuário final.

Está correto o que se afirma em

A) I, apenas.

B) I e II, apenas.

C) I e III, apenas.

D) II, apenas.

E) I, II e III.

25 - FCC – 2013 – MP/RN

Analise as descrições dos tipos de teste:

I. É feito para determinada quantidade de dados ou transações que deveriam ser típicos para um sistema e avalia o comportamento do sistema em termos de tempo para esses dados ou transações. Dessa forma, pode-se verificar se o sistema atende aos requisitos de performance estabelecidos e também se existem gargalos de performance para serem tratados.

II. Procura-se levar o sistema ao limite máximo de funcionamento esperado, para verificar como ele se comporta. É feito para verificar se o sistema é suficientemente robusto em situações anormais de carga de trabalho.

III. É feito para verificar se o sistema consegue manter suas características de performance durante um longo período de tempo com uma carga nominal de trabalho. Deve ser verificado o uso da memória ao longo do tempo para garantir que não existam perdas acumulativas de memória e também se não existe degradação de performance após um substancial período de tempo em que o sistema opera com carga nominal ou acima dela.

A associação correta entre o tipo de teste e a descrição é:

A) Teste caixa-branca – Teste de Estresse – Teste de Segurança
B) Teste de carga – Teste de recuperação de falha – Teste de instalação
C) Teste de recuperação de falha – Teste de segurança – Teste caixa-preta
D) Teste de carga – Teste de estresse – Teste de resistência
E) Teste caixa-branca – Teste de recuperação de falha – Teste de segurança

26 - FCC – 2014 – ALE/PE

Isabel trabalha como Analista Legislativo na Assembleia Legislativa do Estado de Pernambuco e ficou responsável por definir qual tipo de teste seria mais adequado para as situações descritas abaixo.

I. O sistema deve ser resistente a falhas, ou seja, falhas de processamento não devem causar a interrupção da sua função global. O teste deve forçar o software a falhar de diversos modos e verificar se a reabilitação é adequadamente realizada.

II. As informações armazenadas pelo sistema devem ser protegidas de todo o tipo de invasão e ataque. O teste deve tentar invadir o sistema e atacar suas vulnerabilidades de forma a verificar se os mecanismos de proteção são realmente capazes de protegê-lo.

III. O sistema deve ser capaz de suportar grande demanda por recursos.

O teste deve submeter o sistema a situações extremas de demanda por recursos, frequência ou volume anormais. Isabel indicou, de forma adequada e respectiva, os seguintes testes para as situações I, II e III:

A) Recuperação, Segurança e Estresse.

B) Reabilitação, Proteção e Desempenho.

C) Tolerância a Falhas, Segurança e Demanda.

D) Desempenho, Proteção e Exaustão.

E) Tolerância a Falhas, Invasão e Estabilidade.

27 - FCC – 2014 – ALE/PE

Os testes de caixa preta (CP) e os testes de caixa branca (CB) apresentam as seguintes características:

I. Referem-se a testes que são conduzidos na interface do software. Examinam algum aspecto fundamental do sistema, sem se preocupar com a estrutura lógica interna do software.

II. Testes exaustivos podem ser impraticáveis, mas podem ser aplicados testes que examinam caminhos lógicos importantes e estruturas de dados essenciais podem ser submetidas à prova quanto à sua validade.

III. São baseados em um exame rigoroso do detalhe procedimental. Caminhos lógicos internos ao software e colaborações entre componentes são testados, definindo-se casos de teste que exercitam conjuntos específicos de condições e/ou ciclos.

IV. Focalizam os requisitos funcionais do software, permitindo ao engenheiro de testes derivar conjuntos de condições de entrada que vão exercitar plenamente todos os requisitos funcionais de um programa.

V. Tentam encontrar erros: em funções incorretas ou omitidas, de interface, de comportamento ou desempenho, de iniciação e término.

VI. Ao usá-los, o engenheiro de testes pode derivar casos de teste que garantam que todos os caminhos independentes de um módulo tenham sido exercitados pelo menos uma vez.

A associação dos tipos de teste de CP ou testes de CB com as características de I a VI é apresentada, correta e respectivamente, em:

A) CB - CP - CP - CB - CB - CP

B) CP - CB - CB - CB - CP - CP

C) CP - CB - CB - CP - CP - CB

D) CB - CP - CP - CP - CB - CP

E) CB - CB - CP - CB - CP - CB

Gabarito - TESTES

01 – C	10- B	19 - E	
02 – A	11- C	20 - D	
03 – B	12- A	21 - C	
04 – A	13- A	22 - C	
05 – E	14 – D	23 - A	
06- B	15 - C	24 - E	
07- A	16 - C	25 - D	
08- B	17 - A	26 - A	
09- D	18 - E	27 - C	

QUESTÕES – ENG. REQUISITOS

01 - FCC - 2011 - TCE-PR

No processo de engenharia de requisitos, os tipos de requisitos de usuário e de sistema podem ser, respectivamente,

a) apenas funcionais; apenas não funcionais.

b) apenas não funcionais; apenas funcionais.

c) apenas funcionais; funcionais e não funcionais.

d) funcionais e não funcionais; apenas não funcionais.

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

02 - FCC - 2010 - BAHIAGÁS

É uma restrição sobre os serviços ou as funções oferecidos pelo sistema. Pode ser uma restrição de *timing, sobre o processo de desenvolvimento, sobre o desempenho ou sobre a confiabilidade do sistema, entre outras. Trata-se de*

a) requisito não funcional.

b) requisto funcional.

c) especificação de risco.

d) iteração de processo.

e) etnografia.

03 - FCC - 2012 - TCE-AP

Em relação a requisitos de sistemas, considere:

I. O modo como um sistema deve reagir a certas entradas e o comportamento em que o sistema deve ter em certas situações e, em alguns casos, especificar o que o sistema não deve fazer, são chamados de requisitos não-funcionais.

II. As restrições aos serviços ou funções de um sistema, como, por exemplo, processos de desenvolvimento ou utilização de padrões, são requisitos de funcionamento do sistema ou requisitos funcionais.

III. Requisitos que vem do domínio da aplicação do sistema e refletem características ou restrições para aquele domínio são chamados de requisitos de domínio e podem ser requisitos funcionais e/ou não-funcionais.

Está correto o que se afirma em

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

04 - FCC - 2012 - MPE-PE

Os requisitos não funcionais não estão diretamente ligados aos serviços específicos oferecidos pelo sistema a seus usuários. Eles podem estar relacionados às propriedades emergentes do sistema, como confiabilidade, tempo de resposta e ocupação de área, entre outros. Dentre os tipos de requisitos não funcionais, é possível destacar os requisitos de produto, organizacionais e externos. Dentre os requisitos de produto, podemos citar os requisitos

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

05 - FCC - 2009 - SEFAZ-SP

Quanto aos requisitos de *software*, considere:

I. É importante que se estabeleçam práticas para encontrar, documentar, organizar e rastrear os requisitos variáveis de um sistema.
II. Etnografia (observação e análise dos fluxos de trabalho) e sessões de JAD são práticas que podem ser aplicadas na elicitação.
III. Elicitar significa descobrir os requisitos de um sistema por meio de entrevistas, de documentos do sistema existente, de análise do domínio do problema ou de estudos do mercado.

Está correto o que se afirma em

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

06 - FCC - 2013 - DPE-SP

Em uma das etapas da Engenharia de Requisitos há a preocupação em se observar a especificação produzida, visando verificar que os requisitos tenham sido declarados, por exemplo, sem ambiguidades.

O texto refere-se à etapa de

a) gestão dos requisitos.

b) elicitação dos requisitos.

c) negociação dos requisitos.

d) levantamento dos requisitos.

e) validação dos requisitos.

07 - FCC - 2012 - TRE-SP

Enquanto a definição de requisitos para um novo sistema é desenvolvida, uma melhor compreensão da necessidade dos usuários é alcançada, e é esperado que haja uma evolução nos requisitos do sistema para acomodar este novo entendimento das necessidades dos usuários. A partir dessa perspectiva de evolução, os requisitos são divididos em duas classes, permanentes e voláteis. Sobre a divisão dos requisitos voláteis, considere:

I. Requisitos mutáveis surgem à medida que a compreensão do cliente sobre o sistema aumenta, tornando-o apto a sugerir e requisitar mudanças.

II. Requisitos consequentes estão diretamente ligados a introdução de sistemas de computação na empresa, que podem modificar processos e criar novos métodos de trabalho.

III. Requisitos emergentes são os requisitos relativamente estáveis, que derivam da atividade principal da organização e se relacionam diretamente com o domínio do sistema.

Está correto o que consta em

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

08 - FCC - 2009 - TRT

No processo de engenharia de requisitos, é uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais. Trata-se de

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

09 - FCC - 2012 - TJ-PE

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

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

10 - FCC - 2014 – TRF

Considere as seguintes atividades:

1. Compreensão do domínio: os analistas devem desenvolver sua compreensão do domínio da aplicação.

2. Coleta de requisitos: processo de interagir com os stakeholders do sistema para descobrir seus requisitos.

3. Classificação: atividade que considera o conjunto não estruturado dos requisitos e os organiza em grupos coerentes.

4. Resolução de conflitos: Solucionar conflitos decorrentes do envolvimento de múltiplos stakeholders.

5. Definição das prioridades: envolve a interação com os stakeholders para a definição dos requisitos mais importantes.

6. Descarte de requisitos: atividade de descartar requisitos menos importantes, baseando-se nas indicações dos stakeholders.

7. Verificação de requisitos: os requisitos são verificados para descobrir se estão completos e consistentes e se estão em concordância com o que os stakeholders desejam do sistema.

8. Modelagem de requisitos: os requisitos são modelados utilizando-se o diagrama de casos de uso e de sequência da UML.

Faz parte do processo de levantamento e análise de requisitos o que consta em APENAS 1, 2,

(A) 3, 4, 5, 7 e 8.

(B) 3, 4, 5, 6.

(C) 3, 4, 5 e 7.

(D) 4, 5, 7 e 8.

(E) 3, 4, 6 e 8.

10 - FCC - 2014 – TRF

Levantamento e Análise de Acordo com Sommerville:

1. Compreensão do domínio: Os analistas devem desenvolver sua compreensão do domínio da aplicação;
2. Coleta de requisitos: É o processo de interagir com os stakeholders do sistema para descobrir seus requisitos. A compreensão do domínio se desenvolve mais durante essa atividade;
3. Classificação: Essa atividade considera o conjunto não estruturado dos requisitos e os organiza em grupos coerentes;
4. Resolução de conflitos: Quando múltiplos stakeholders estão envolvidos, os requisitos apresentarão conflitos. Essa atividade tem por objetivo solucionar esses conflitos;
5. Definição das prioridades: Em qualquer conjunto de requisitos, alguns serão mais importantes do que outros. Esse estágio envolve interação com os stakeholders para a definição dos requisitos mais importantes;
6. Verificação de requisitos: Os requisitos são verificados para descobrir se estão completos e consistentes e se estão em concordância com o que os stakeholders desejam do sistema.

11 - FCC - 2014 – SABESP

Após um estudo inicial de viabilidade, o próximo estágio do processo de engenharia de requisitos é a elicitação e análise de requisitos. Nesta atividade deve-se

(A) permitir que os engenheiros de software trabalhem com os clientes e os usuários finais para obterem apenas informações sobre o domínio da aplicação e os serviços que o sistema pode oferecer.

(B) interagir com os stakeholders por meio da observação e de entrevistas. Cenários e protótipos podem ser utilizados para ajudar os stakeholders a compreenderem o que o sistema vai incorporar.

(C) considerar apenas os requisitos vindos dos stakeholders, descartando-se os requisitos provenientes de todos os outros sistemas.

(D) realizar entrevistas, que são a melhor técnica para compreender os requisitos do domínio da aplicação, pois o conheci- mento de domínio é tão familiar aos stakeholders que eles têm facilidade de explicá-lo.

(E) usar entrevistas, que são uma técnica eficaz para a elicitação do conhecimento sobre os requisitos e restrições organiza- cionais porque as estruturas organizacionais que serão descritas corresponderão fielmente à realidade do processo de tomada de decisão.

12 - FCC - 2011 - TRT

Tabelas de rastreamento para relacionar os requisitos identificados a um ou mais aspectos do sistema ou do seu ambiente devem ser desenvolvidas, segundo Pressman, na engenharia de requisitos por meio da função de

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

13 - FCC - 2013 – Defensoria SP

A prototipação representa uma técnica poderosa para o desenvolvimento de sistemas, mais especificamente do software desses sistemas. Sobre as funções desempenhadas por um protótipo, é correto afirmar que ele

(A) permite avaliar o desempenho geral da equipe de desenvolvimento de software.

(B) não permite que sejam realizados testes, visando verificar o funcionamento do sistema final, ainda que sejam testes parciais.

(C) é inteiramente descartado, não sendo aproveitada nenhuma parte do código de software no sistema final entregue ao cliente

(D) não possibilita avaliar a qualidade do software produzido.

(E) pode auxiliar na validação de requisitos do sistema, bem como propiciar a inserção de novos requisitos ainda não identificados.

14 - FCC - 2010 - MPE-RN

Sobre requisito funcional, considere:

I. O sistema deve fornecer telas apropriadas para o usuário ler os documentos no repositório de documentos.

II. O usuário deve ser capaz de fazer uma busca em todo o conjunto inicial de banco de dados.

III. O sistema deve atender aos requisitos de confiabilidade, usabilidade e portabilidade.

Está correto o que se afirma em

a) I, apenas.

b) II, apenas.

c) III, apenas.

d) I e II, apenas.

e) I, II e III.

15 - FCC - 2010 - MPE-RN

As políticas de rastreabilidade de requisitos são decididas durante o estágio de

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

16 - FCC - 2010 - MPE-RN

Na engenharia de software, etnografia é

a) uma fase do processo de software aplicada no modelo em cascata.
b) uma fase do processo de software aplicada no modelo em espiral.
c) uma técnica de observação que pode ser usada para compreender os requisitos sociais e organizacionais.
d) uma técnica aplicada na engenharia de requisitos cujo objetivo é definir, a priori, as classes que contém elementos gráficos (BLOB).
e) um projeto cujo principal objetivo é criar interfaces gráficas, que facilitam o acesso do usuário (GUI).

17 - FCC - 2009 - TRT

Dentre os requisitos não funcionais, classificados em

I. De produto.
II. Organizacionais.
III. Externos.

Corresponde a I, II e III, respectivamente,

a) segurança; privacidade; desempenho.
b) interoperabilidade; usabilidade; desempenho.
c) interoperabilidade; desempenho; ético.
d) portabilidade; de entrega; interoperabilidade.
e) usabilidade; de segurança; de privacidade.

18 - FCC - 2009 - TRT

Com relação aos requisitos de software, considere:

I. funcionais são somente requisitos de usuário.

II. funcionais e não-funcionais podem ser requisitos de usuário.

III. funcionais e não-funcionais podem ser requisitos de sistema.

Está correto o que se afirma APENAS em

a) I.

b) II.

c) III.

d) I e III.

e) II e III.

19 - FCC - 2009 - PGE-RJ

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.

20 - FCC - 2012 - SABESP

A descoberta de requisitos do sistema é o processo de reunir informações sobre o sistema requerido e sobre sistemas existentes. Sobre essa fase, considere:

I. Diagramas de Casos de Uso são utilizados na fase de descoberta de requisitos e identificam as interações individuais entre o sistema e seus usuários ou outros sistemas.

II. Os cenários podem ser particularmente úteis para adicionar detalhes a uma descrição geral de requisitos. Cada cenário geralmente cobre um pequeno número de interações possíveis.

III. Durante as entrevistas com os envolvidos no sistema (stakeholders), a equipe responsável pelo levantamento de requisitos levanta questões sobre o sistema atual. Essas entrevistas podem ser de dois tipos: fechadas ou abertas.

Está correto o que consta em

(A) II, apenas.

(B) I e II, apenas.

(C) I e III, apenas.

(D) II e III, apenas.

(E) I, II e III.

21 - FCC - 2012 – METRO/SP

Os requisitos não funcionais surgem por meio das necessidades dos usuários, como restrições de orçamento, políticas organizacionais ou mesmo por fatores externos, como regulamentos de segurança e legislações de privacidade. Dentre a classificação dos requisitos não funcionais estão os requisitos de produto, os quais

(A) especificam ou restringem o comportamento do software, incluindo requisitos de desempenho, especificações de rapidez de execução e requisitos de confiabilidade que estabelecem, por exemplo, a taxa aceitável de falhas.

(B) são os requisitos gerais de sistemas derivados das políticas e procedimentos da organização do cliente e do desenvolvedor, como, por exemplo, os requisitos de processo operacional.

(C) definem os requisitos do processo de desenvolvimento, como, por exemplo, a linguagem de programação, o ambiente de desenvolvimento ou normas do processo a serem usadas.

(D) abrangem todos os requisitos que derivam de fatores externos ao sistema e seu processo de desenvolvimento. Podem incluir requisitos reguladores, que definem o que deve ser feito para que o sistema seja aprovado para uso.

(E) incluem os requisitos legais, os quais devem ser seguidos para garantir que o sistema opere dentro da lei, e os requisitos éticos, os quais asseguram que o sistema será aceitável para seus usuários e o público geral.

22 - FCC - 2012 – TER/CE

Considere:

I. Para cada cliente deve ser aplicado um identificador único.

II. O tempo de resposta entre a requisição e a informação não pode exceder a 2 ms.

III. Clientes têm filiais que devem "carregar", na base de dados, o identificador do cliente principal.

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

São requisitos não funcionais os que constam em

(A) I e II, apenas.

(B) II e III, apenas.

(C) II e IV, apenas.

(D) I, III e IV, apenas.

(E) I, II, III e IV.

23 - FCC - 2012 – TRE/RN

Considere os requisitos:

I. Os valores das faturas devem ser totalizados por cliente e por data de vencimento igual à fornecida pela área de contas a pagar.

II. O software deve ser processável tanto em alta quanto em baixa plataforma.

III. A data de vencimento constante dos boletos de pagamento deve ser igual à data de registro de entrada do documento no cadastro, mais 30 dias corridos.

Exemplo de requisito não funcional consta APENAS em

(A) I.

(B) II.

(C) III.

(D) I e II.

(E) II e III.

24 - FCC - 2011 – TRT1/RJ

A técnica utilizada na compreensão de requisitos sociais e organizacionais por observação das rotinas dos envolvidos é a:

(A) prototipação.
(B) por pontos de vista.
(C) por cenário.
(D) entrevista.
(E) etnografia.

25 - FCC - 2011 – TRT19/RJ

A avaliação do impacto de mudança de um requisito, muitas vezes, faz com que seja necessário retornar à sua fonte. Na validação dos requisitos, a equipe deve estar atenta, portanto, à

(A) rastreabilidade.

(B) adaptabilidade.

(C) qualidade.

(D) facilidade de compreensão.

(E) facilidade de verificação.

26 - FCC - 2011 – TRT19/RJ

De acordo com *Sommerville, são atividades do processo de elicitação de requisitos, pela ordem:*

(A) casos de uso; análise; projeto; arquitetura.

(B) etnografia; casos de uso; análise; validação; arquitetura.

(C) entrevista; etnografia; documentação; registro.

(D) cenários; classificação; organização; priorização; documentação.

(E) obtenção; classificação e organização; priorização e negociação; documentação.

27 - FCC - 2012 – TST (Adaptada)

Na Engenharia de Requisitos, o gerente de requisitos classifica os requisitos em diferentes tipos, sendo os do tipo funcional relacionados com o custo e confiabilidade do software e os do tipo não-funcional relacionados com os casos de uso.

Certo Errado

28 - FCC – 2011 – INFRAERO (Adaptada)

No contexto de levantamento de requisitos, funcionalidade é um dos aspectos que deve ser levado em conta na abordagem dos requisitos funcionais.

Certo Errado

29 - FCC – 2010 – TRT8(Adaptada)

É considerado um requisito não funcional o tempo de resposta máximo.

Certo Errado

30 - FCC – 2010 – BAHIAGAS(Adaptada)

É uma restrição sobre os serviços ou as funções oferecidas pelo sistema. É uma restrição sobre os serviços ou as funções oferecidas pelo sistema, entre outras. Trata-se de requisitos funcional

Certo Errado

31 - FCC – 2009 – SEFAZ/SP

É necessário que o software calcule os salários dos diaristas e mensalistas e emita relatórios mensais sumariados por tipo de salário. Entretanto, a base de dados deve estar protegida e com acesso restrito aos usuários autorizados. De qualquer forma, o tempo de resposta das consultas não deve superar os quinze segundos, pois inviabilizaria todo o investimento nesse sistema. Devo lembrar que os relatórios individuais dos departamentos, nos quais constam os salários dos funcionários, devem ser emitidos quinzenalmente em razão dos adiantamentos e vales que recebem. É fundamental que o software seja operacionalizado usando código aberto.
Necessito, ainda, forte gerenciamento de risco, prazo e custo, porque a entrega do produto final não pode ultrapassar o prazo de oito meses a contar da data de início do projeto. No texto, são requisitos funcionais:

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

32 - FCC – 2010 – DPE/SP

No contexto da Engenharia de Requisitos, considere:

I. O sistema deve fornecer uma entrada de dados que possibilite a inclusão de atributos de permissão de acesso às dependências da corporação por técnicos, supervisores e chefes.
II. Algumas permissões de acesso deverão ter tratamento especial para a entrada de atributos. Para este tipo de permissão, atributos excedentes a uma faixa predeterminada só poderão ser incluídos por chefes de seção.

Em relação às assertivas acima, é correto afirmar:

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

33 - FCC – 2010 – SEFAZ/SP

Dentre os requisitos obtidos para a construção do software constavam:
1. O software deve permitir as funções de cadastro, consultas diversas, alteração de dados e exclusão de alunos, professores e demais colaboradores.
2. O sistema deve ser fácil de usar, fácil de encontrar o que se procura e fácil de memorizar os passos para executar as operações mais comuns.
3. O sistema deve ter seu funcionamento baseado nas tecnologias web.
4. Todas as operações disponibilizadas no sistema devem contemplar a legislação vigente.
5. O sistema deve fazer interface com o sistema da Receita Federal por meio de requisições/respostas utilizando XML.
6. Os alunos devem poder obter por meio do sistema informações sobre suas faltas e notas em cada disciplina.
7. O boletim e o histórico do aluno poderão ser consultados e visualizados pelos gestores, funcionários da secretaria e pelo próprio aluno.
8. Ao clicar em uma opção para gerar o boletim do aluno, deve ser apresentada ao solicitante uma tabela com todas as disciplinas que o aluno cursou, bem como as notas das provas e o número total de faltas em cada disciplina.
9. O sistema deve responder à solicitação de geração do boletim de um aluno em no máximo 10 segundos.
10. O sistema deve calcular a média aritmética das duas maiores dentre três notas de cada disciplina no final do semestre.
11. Quando o sistema constatar que o aluno tem mais que 25% de faltas em uma disciplina do semestre, deve ser exibida no boletim do aluno a informação "Reprovado".
12. O sistema deverá suportar a execução em qualquer plataforma de hardware e/ou sistema operacional.
13. O sistema deve enviar automaticamente para o e-mail dos gestores autorizados um relatório com o número de alunos inadimplentes por curso.
14. O sistema não deve revelar quaisquer dados pessoais dos alunos aos professores, exceto informações sobre notas e faltas no curso em que o professor leciona.
15. O sistema deve permitir que o professor inclua ou modifique as notas de seus alunos durante o semestre letivo.
16. A quantidade de memória necessária para que um terminal possa executar o sistema nas condições mínimas aceitáveis é de 1 gigabyte.
17. A taxa aceitável de falhas nas operações realizadas pelo usuário no sistema deve ser de 1 falha para cada 200 operações.
18. O sistema e sua respectiva documentação deverão ser entregues em um ano a partir da data atual.
19. O sistema não deve permitir operações que beneficiem alguns usuários em detrimento de outros.
20. A interface do usuário deve ser construída utilizando HTML5 e CSS.
21. Se a média do aluno por disciplina, calculada no final do semestre, for menor do que 7, deve ser exibido no boletim do aluno a informação "Reprovado".

Baseado nos requisitos apresentados, é correto afirmar que são requisitos funcionais os de números:

a) 1, 2, 6, 10, 11, 14, 15, 16 e 21.

b) 1, 6, 8, 10, 11, 13, 14, 17, 18 e 19.

c) 1, 6, 7, 8, 10, 11, 13, 15 e 21.

d) 1, 3, 4, 8, 10, 11, 12, 13, 15, 18 e 21.

e) 2, 3, 4, 5, 9, 12, 14, 16, 17, 18, 19 e 20.

34 - FCC – 2013 – MPE/MA

O escopo de um projeto é determinado pelo levantamento de requisitos funcionais e não funcionais. Dentre os requisitos não funcionais se enquadram os requisitos organizacionais, que podem ser divididos em:

a) reguladores e éticos.

b) ambientais, operacionais e de desenvolvimento.

c) contábeis e de segurança.

d) de desempenho e de espaço.

e) de eficiência, de confiança e de proteção.

35 - FCC – 2011 – TRT23

Detalhes técnicos desnecessários especificados pelos usuários podem confundir os objetivos globais do sistema. No levantamento de requisitos, trata-se de um problema de

a) Complexidade

b) Detalhamento

c) Entendimento

d) Escopo

e) Volatilidade

36 - FCC – 2012 – PMSP

- Os requisitos não funcionais podem ser divididos em 3 grupos: requisitos de produto, requisitos organizacionais e requisitos externos. Dentre os grupos de requisitos externos se encontram os requisitos

a) Ambientais

b) De proteção

c) Reguladores

d) Operacionais

e) De desempenho

37 - FCC – 2010 – MPE/SP

Na Engenharia de Software, no âmbito da atividade de levantamento de requisitos, duas abordagens são consideradas: os requisitos funcionais e os requisitos não-funcionais.

É um exemplo típico de requisito funcional:

a) Facilidade de manutenção

b) Segurança

c) Facilidade de uso

d) Funcionalidae

e) Desempenho

38 - FCC – 2010 – MPE/SP

A frase "o tempo médio de resposta do sistema não deve ultrapassar 5 segundos" indica

a) uma funcionalidade do sistema.

b) uma atividade do cronograma do sistema.

c) uma função executada pelo usuário do sistema.

d) uma possível definição de requisito não funcional.

e) um ponto de controle nas etapas de desenvolvimento do sistema.

39 - FCC – 2010 – TRE/RN

Um analista se insere no ambiente de trabalho onde o sistema será usado. Ele observa o trabalho rotineiro e anota as tarefas reais nas quais os participantes estão envolvidos. Trata-se da técnica de elicitação e análise de requisitos denominada

a) Workshop.
b) Casos de uso.
c) Validação de requisitos.
d) Etnografia.
e) Entrevista.

40 - FCC – 2010 – MPE/SE

Os requisitos não funcionais surgem por meio das necessidades dos usuários, devido a restrições de orçamento, políticas organizacionais, necessidade de interoperabilidade e fatores externos. Estes requisitos podem ser classificados como requisitos de produto, organizacionais e externos. Os requisitos externos ainda são classificados como reguladores, éticos e

A) De desempenho
B) De proteção
C) Ambientais
D) De usabilidade
E) Legais

Gabarito – ENG. REQUISITOS

01 - E	09 – D	17 – D	26 – A	34 - B
02 – A	10 – C	18 – E	27 – ERRADO	35 - D
03 – A	11 – B	19 – E	28 – CERTO	36 - C
04 – A	12 – A	20 – E	29 - CERTO	37 - D
05 – C	13 – E	21 – A	30 - ERRADO	38 - D
06 – E	14 – D	22 – C	31 - A	39 - D
07 – A	15– E	23 – B	32 - A	40 - E
08 – E	16 – C	25 – E	33 - C	

APF

06 – FCC - 2012 – TJ/PE

Considere:

I. Contagem de pf detalhada.
II. Contagem de pf estimativa.
III. Contagem de pf indicativa.

Quanto ao tipo de contagem, a Netherlands Software Metrics Association reconhece o que consta em

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

07 – FCC - 2012 – TRT11

Segundo a IFPUG em relação à métrica do software por análise por pontos de função, considere:

I. Análise por pontos de função executa a medição do software determinando a quantidade de funcionalidades que o software fornece ao usuário baseado principalmente na arquitetura lógica.

II. O objetivo da análise por pontos de função é medir as funcionalidades que o usuário requisita e recebe e, também, medir o desenvolvimento e manutenção do software com dependência na implementação utilizada pela empresa.

III. O processo de contagem dos pontos de função deve ser simples o suficiente para minimizar a sobrecarga do processo de medida e consistente dentre os vários projetos e organizações.

Está correto o que se afirma em:

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

08 – FCC - 2011 – TRT1

No processo de Análise de Pontos de Função - APF, aplicam-se os mesmos valores: 3, 4 e 6, correspondentes, respectivamente, aos níveis simples, médio e complexo, nos tipos de função:

a) entrada externa e saída externa.

b) entrada externa e consulta externa.

c) consulta externa e saída externa.

d) arquivo lógico interno e consulta externa.

e) arquivo lógico interno e arquivo de interface externa.

09 – FCC - 2012 – TER/CE

Considere 3 AIEs simples, 5 EEs médias, 8 CEs complexas, 3 ALIs complexos e 7 SEs médias. O cálculo de PF bruto é

a) 136.
b) 148.
c) 159.
d) 163.
e) 212.

10 – FCC - 2011 – INFRAERO

Analise a tabela utilizada no cálculo de Pontos de Função:

Tipo de Função	Complexidade Funcional		
	Simples	Média	Complexa
I	7	10	15
II	5	7	10
III	4	5	7

Preenchem correta e respectivamente os tipos de função **I, II** e **III:**

a) ALI, AIE e SE.
b) ALI, CE e AIE.
c) CE, EE e ALI.
d) AIE, AL e EE.
e) EE, CE e SE.

11 – FCC - 2011 – TRT24

Após a aplicação do fator de ajuste, o total de pontos de função em uma contagem ficou em 110,60. Antes da aplicação do ajuste, os pontos de

função brutos estavam em 140,00. Portanto, o somatório dos 14 itens do nível de influência global foi:

a) 11.

b) 14.

c) 15.

d) 18.

e) 19. 98

12 – FCC - 2009 – TCE/GO

Na aplicação da métrica Análise de Pontos por Função, caso haja influência forte em quatro das 14 Características Gerais de Sistema, os pontos ajustados serão

a) 65% dos pontos brutos.

b) 75% dos pontos brutos.

c) 80% dos pontos brutos.

d) 85% dos pontos brutos.

e) 115% dos pontos brutos.

13 – FCC - 2010 – TCM/PA

Quanto aos pontos brutos, na Análise de Pontos de Função o fator de ajuste aplicado pode aumentá-los

a) em até 35% ou diminuí-los em até 65%.

b) ou diminuí-los em até 35%.

c) ou diminuí-los em até 65%.

d) ou diminuí-los em até 1,35%.

e) em até 65% ou diminuí-los em até 35%.

14 – FCC - 2012 – MPE/AP

Dentre os métodos disponíveis na utilização de métricas de sistema está a análise de pontos de função (*Function Point Analysis*). Nesse método,

a) a função realizada pelos objetos do sistema, seus atributos e operações são catalogados, possibilitando medir a quantidade de classes e objetos que serão necessários para este sistema.

b) as funções utilizadas em linguagens de desenvolvimento tradicional, bem como os métodos e operações utilizados em arquiteturas orientadas a objeto são contados para a definição do tamanho funcional do sistema.

c) é atribuída uma pontuação para cada função ou método executado por uma determinada linguagem de programação. Este número é formulado com base em cálculos matemáticos e, posteriormente, é utilizado para fazer a classificação das métricas do sistema.

d) são analisados os pontos de execução de cada função dentro de um determinado sistema, são gerados registros de sistemas (logs) e, posteriormente, é gerada uma classificação em função dos valores obtidos dessa análise. .

e) as funcionalidades do sistema são elencadas sem a necessidade de preocupação com a tecnologia que será utilizada para o desenvolvimento do sistema.

15 – FCC - 2011 –INFRAERO

A métrica análise por pontos de função foi desenvolvida na década de 1970, como uma forma de medir *software*. Analise os itens a seguir relacionados a essa métrica:

I. Considera mais importante o número de linhas de código do que as funcionalidades criadas.

II. Pode ser aplicada antes do código ser escrito, baseando-se na descrição arquitetural do projeto.

III. É dependente da tecnologia utilizada no desenvolvimento.

IV. Dois programas muito diferentes podem possuir a mesma contagem de pontos de função.

Está correto o que consta em

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

16 – FCC - 2011 –TRT14

Analise a tabela utilizada na medição de pontos de função:

Contribuição	Classificação		
	Simples	Média	Complexa
I	5	7	10
II	7	10	15
III	4	5	7

Considerando os valores de Classificação (Simples, Média e Complexa) e as siglas:

- ALI = Arquivo Lógico Interno.

- AIE = Arquivo de Interface Externa.

- SE = Saída Externa.

I, II e III correspondem, respectivamente, às funções

a) SE, ALI e AIE.
b) SE, AIE e ALI.
c) AIE, ALI e SE.
d) AIE, SE e ALI.
e) ALI, AIE e SE.

17 – FCC - 2011 – TRT1

No processo de Análise de Pontos de Função - APF, aplicam-se os mesmos valores: 3, 4 e 6, correspondentes, respectivamente, aos níveis simples, médio e complexo, nos tipos de função:

a) entrada externa e saída externa.

b) entrada externa e consulta externa.

c) consulta externa e saída externa.

d) arquivo lógico interno e consulta externa.

e) arquivo lógico interno e arquivo de interface externa.

18 – FCC - 2011 – TRT4

Considere a tabela a seguir:

Função	Classificação		
	Simples	Média	Complexa
ALI	7	10	X
AIE	5	Y	10
EE	3	4	6
SE	Z	5	7
CE	3	4	6

Na medição de pontos de função, os valores X, Y e Z utilizados na tabela, correspondem, respectivamente, a

a) 12, 7 e 3.
b) 13, 8 e 4.
c) 13, 6 e 4.
d) 15, 8 e 3.
e) 15, 7 e 4.

19 – FCC - 2011 – TER/RN

Funções	Pesos das funções (Simples / Média / Complexa)					
	Peso Simples	Qtde.	Peso Média	Qtde.	Peso Complexa	Qtde.
ALI – Arquivo Lógico Interno	7	2	10	1	15	0
AIE – Arquivo de Interface Externa	5	0	7	0	10	1
EE – Entrada Externa	3	0	4	3	6	0
SE – Saída Externa	4	2	5	0	7	1
CE – Consulta Externa	3	0	4	1	6	0

Características Gerais para cálculo do Fator de Ajuste	Pontos
Comunicação de dados	3
Funções / processamento distribuído	2
Objetivos de desempenho / Performance	0
Configuração de equipamento / Ambiente operacional	5
Volume de transações	0
Entrada de dados *online*	0
Interface com o usuário	0
Atualizações *online*	5
Processamento complexo	0
Código reutilizável	3
Facilidade de implantação / Conversão e instalação	5
Facilidade operacional / *Backup*	2
Múltiplos locais / Portabilidade	0
Facilidade de mudanças (flexibilidade) / Manutenibilidade	5

Após a aplicação do fator de ajuste sobre os pontos de função brutos, obter-se-á

a) 52,25.
b) 57,95.
c) 61,75.
d) 66,50.
e) 67,45.

20 – FCC - 2011 – TRT24

Após a aplicação do fator de ajuste, o total de pontos de função em uma contagem ficou em 110,60. Antes da aplicação do ajuste, os pontos de função brutos estavam em 140,00. Portanto, o somatório dos 14 itens do nível de influência global foi

a) 11.
b) 14.
c) 15.
d) 18.
e) 19.

21 - FCC - 2014 – TRF

Sabendo que a Análise de Pontos de Função (APF) permite medir o tamanho funcional do software, considere que no desen- volvimento de um software foram fornecidos os seguintes dados:

Tabela 1: Complexidade para Entradas Externas (EEs)

AR \ TD	< 5	5-15	>15
< 2	Baixa	Baixa	Média
2	Baixa	Média	Alta
> 2	Média	Alta	Alta

Tabela 2: Complexidade para Saídas Externas (SEs) e Consultas Externas (CEs)

AR \ TD	< 6	6-19	>19
< 2	Baixa	Baixa	Média
2 – 3	Baixa	Média	Alta
> 3	Média	Alta	Alta

Tabela 3: Contribuições dos pontos de função das funções do tipo transação

Tipo de função	Baixa	Média	Alta
Entrada Externa (EE)	3 PF	4 PF	6 PF
Saída Externa (SE)	4 PF	5 PF	7 PF
Consulta Externa (CE)	3 PF	4 PF	6 PF

Tabela 4: Contagem das transações do sistema

Descrição	Tipo	TD	AR	Complexidade	Contribuição
Login no sistema	CE	4	2		
Cadastro de notas	EE	5	3		
Alteração de notas	EE	7	2		
Emitir boletim do aluno	SE	16	5		
Exibir lista de presença	CE	9	2		
Exibir diário de classe	SE	20	4		
Total de pontos de função das transações					

Legenda: AR – Quantidade de Arquivos Referenciados
TD – Quantidades de tipos de dados

Ao se completar a tabela 4, o total de pontos de função das transações é

(A) 35.
(B) 33.
(C) 31.
(D) 28.
(E) 30.

22 - FCC - 2014 – TRT 15ªRegião

A Análise de Pontos de Função (APF) é usada para medir o tamanho funcional do software. Considere que, no desenvolvimento de um software, foram fornecidos os dados abaixo.

Tabela 1: Complexidade funcional dos Arquivos Lógicos Internos (ALI) e Arquivos de Interface Externa (AIE)

		Tipos de Dados (TD)		
		< 20	20 - 50	> 50
Tipos de Registros (TR)	1	Baixa	Baixa	Média
	2 - 5	Média	Média	Alta
	> 5	Média	Alta	Alta

Tabela 2: Contribuição dos Pontos de Função (PF) por tipo de função

Tipo de Função	Baixa	Média	Alta
Arquivo Lógico Interno (ALI)	7 PF	10 PF	15 PF
Arquivo de Interface Externa (AIE)	5 PF	7 PF	10 PF
Entrada Externa (EE)	3 PF	4 PF	6 PF
Saída Externa (SE)	4 PF	5 PF	7 PF
Consulta Externa	3 PF	4 PF	6 PF

O fragmento de tabela abaixo foi construído para fazer a contagem de Pontos de Função de um projeto de desenvolvimento de software.

Nome da Função	Tipo	TD	TR	Complexidade	Contribuição
Função A	AIE	3	1	Baixa	I .....
Função B	SE	21	6	II	7 PF
Função C	ALI	3	2	Média	III .....
Função D	EE	51	7	IV	6 PF

Com base nos dados apresentados, pode-se afirmar que as lacunas I, II, III e IV são preenchidas correta e, respectivamente, com:

(A) 5 PF, Alta, 10 PF, Alta.
(B) 5 PF, Média, 15 PF, Média.
(C) 7 PF, Média, 7 PF, Média.
(D) 5 PF, Alta, 10 PF, Média.
(E) 7 PF, Média, 15 PF, Alta.

23 - FCC - 2012 – BANESE

Considere a tabela:

Função	Quantidades		
	Simples	Média	Complexa
ALI	1	1	1
AIE	1	1	1
EE	1	1	1
SE	1	1	1
CE	1	1	1

Após o cálculo de pontos de função brutos, será obtido o valor

(A) 96.

(B) 99.

(C) 106.

(D) 109.

(E) 116.

24 - FCC - 2010 – TRT9

Na análise de pontos de função, são apenas do tipo Transação as funções

A) ALI e SE.

B) ALI e AIE.

C) ALI, AIE e SE.

D) CE, EE e SE.

E) CE, EE, SE e AIE.

25 - FCC - 2011 – TRE/AP

Na métrica de pontos de função, Entrada Externa de média complexidade e Arquivo Lógico Interno de alta complexidade valem, respectivamente, em pontos

A) 3 e 7.

B) 3 e 10.

C) 4 e 10.

D) 4 e 15.

E) 5 e 15.

26 - FCC - 2011 – TCE/SE

Uma contagem APF onde existem 10 ALI simples, 5 EE médias, 2 SE complexas e 5 CE complexas, sem aplicação do fator de ajuste, resultará em

A) 118 pontos.

B) 126 pontos.

C) 128 pontos.

D) 134 pontos.

E) 162 pontos.

27 - FCC - 2011 – TCE/AM

NÃO é uma das nove características em um tratamento detalhado das métricas de software (Whitmire) para o modelo de projeto orientado a objeto:

A) o custo.

B) o tamanho.

C) a complexidade.

D) o acoplamento.

E) a coesão.

28 - FCC - 2008 – TCE/AL

Durante a medição do grau de complexidade de um sistema foram apurados 550 pontos de função brutos. Considerando que o somatório dos graus atribuídos aos fatores de ajuste foi 30, a medida final em pontos de função foi

A) 520

B) 522,5

C) 552,5

D) 580

E) 585,5

29 - FCC - 2009 – INFRAERO

O valor do fator de ajuste poderá ajustar os pontos de função (PF) não ajustados em

A) até 35% a mais, apenas.

B) até 35% a menos ou em até 35% a mais.

C) torno de 35% a menos, apenas.

D) torno de 35% a mais, apenas.

E) torno de 35% a menos ou em torno de 35% a mais.

30 - FCC - 2009 – INFRAERO

O tipo de contagem de pontos de função (PF) denominado Saídas Externas representa a quantidade de

A) arquivos lógicos atualizados no sistema.

B) arquivos lógicos atualizados fora do sistema.

C) processos que apenas recuperam dados e os enviam para fora do sistema.

D) processos que envolvem cálculos e enviam os dados para fora do sistema.

E) arquivos de interface externa.

31 - FCC - 2010 – INFRAERO

Tipo que NÃO pertence ao domínio de informação da métrica Ponto de Função (FP - Function Point):

A) número de entradas externas (external inputs Eis).

B) número de saídas externas (external outputs EOs).

C) número de consultas externas (external inquiries EQs).

D) número de transações lógicas externas (external logical transactions ELTs).

E) número de arquivos de interface externa (external interface files EIFs).

32 - FCC - 2013 – MPE/SE

Uma das utilizações mais comuns da Análise de Pontos de Função (APF), no Brasil, tem sido para

A) medir como e em qual linguagem o software é construído.

B) medição de contratos de software.

C) medir o custo do projeto baseado nos requisitos funcionais e não funcionais e na tecnologia utilizada.

D) medição de desempenho, usabilidade e portabilidade do software.

E) medir o tamanho do software com base nos programas e em suas funções ou métodos e nas linhas de código utilizadas para cada funcionalidade.

33 - FCC - 2009 – TRT/PI

O valor do fator de ajuste (VAF) para calcular os pontos de função ajustados é baseado

A) nas características gerais de sistema e no nível de influência de cada característica.

B) nas características gerais de sistema e na complexidade das funções do sistema.

C) nas funções do sistema e no nível de influência de cada função.

D) nos pontos de função não ajustados e no nível de influência das características do sistema.

E) nos pontos de função não ajustados e na complexidade de cada função do sistema.

34 - FCC - 2013 – MP/MA

A Análise por Pontos de Função é uma técnica paramétrica para estimar o esforço para o desenvolvimento de software. Sobre esta técnica pode-se afirmar que

A) é aplicável apenas após os programas terem sido criados.

B) não se baseia em requisitos, mas em linhas de código.

C) as medidas obtidas por esta técnica são sempre dependentes da linguagem de programação e da tecnologia empregada.

D) todo e qualquer requisito conta como função.

E) pode ser aplicada para medir o tamanho de um sistema antes de desenvolvê-lo.

35 - FCC - 2010 – TRF4

Sobre a métrica análise por pontos de função, é correto afirmar:

A) A medida não pode ser aplicada com base na descrição arquitetural do projeto, mas sim no código desenvolvido.

B) É dependente da tecnologia utilizada no desenvolvimento.

C) A contagem de pontos de função pode ser aplicada logo após a definição da arquitetura, permitindo estimar o esforço e o cronograma de implementação de um projeto.

D) Para determinar o número de pontos de função, deve-se desconsiderar a contagem de dados e de transações.

E) Não pode ser aplicada para estimar esforço de manutenção em sistemas já em funcionamento.

36 - FCC - 2013 – ALE/RN

A técnica de Análise por Pontos de Função - APF

A) deve ser utilizada para estimar a complexidade ciclomática dos programas de computador baseado em suas funções.

B) pode ser aplicada para medir o tamanho de um sistema antes de desenvolvê-lo, de forma que seu custo seja previsto mais adequadamente.

C) é aplicável antes mesmo dos requisitos funcionais do software serem definidos.

D) é baseada no número de linhas de código produzidas, sendo mais adequada para medir a produtividade da equipe de programadores.

E) permite a contagem de pontos de função somente para estimar o esforço de desenvolvimento de novos projetos.

37 - FCC - 2012 – MPE/AP

Dentre os métodos disponíveis na utilização de métricas de sistema está a análise de pontos de função (Function Point Analysis). Nesse método,

A) a função realizada pelos objetos do sistema, seus atributos e operações são catalogados, possibilitando medir a quantidade de classes e objetos que serão necessários para este sistema.

B) as funções utilizadas em linguagens de desenvolvimento tradicional, bem como os métodos e operações utilizados em arquiteturas orientadas a objeto são contados para a definição do tamanho funcional do sistema.

C) é atribuída uma pontuação para cada função ou método executado por uma determinada linguagem de programação. Este número é formulado com base em cálculos matemáticos e, posteriormente, é utilizado para fazer a classificação das métricas do sistema.

D) são analisados os pontos de execução de cada função dentro de um determinado sistema, são gerados registros de sistemas (logs) e, posteriormente, é gerada uma classificação em função dos valores obtidos dessa análise.

E) as funcionalidades do sistema são elencadas sem a necessidade de preocupação com a tecnologia que será utilizada para o desenvolvimento do sistema.

38 - FCC - 2012 – TST

O Gerente de Projetos de Software aplica os conhecimentos, habilidades e ferramentas às atividades do projeto com o objetivo de garantir que o produto seja desenvolvido de acordo com os requisitos. A métrica de análise de Pontos de Função, de acordo com a norma ISO/IEC 20968,

A) auxilia o Gerente de Projetos de Software a estimar o esforço necessário e custo para o desenvolvimento de sistemas com a abordagem da análise estruturada de sistemas.

B) possibilita ao Gerente de Projetos de Software medir o esforço e qualidade necessários para desenvolver softwares, desde que esteja usando a análise orientada a objetos e os diagramas da UML.

C) classifica a contagem das funções do tipo dado em Entradas Externas (EE), Consultas Externas (CE) e Saídas Externas (SE), representando requisitos que gerem o armazenamento de dados do usuário.

D) define que os Arquivos Lógicos Internos (ALI) e os Arquivos de Interface Externa (AIE) são funções do tipo transição, os quais representam requisitos relacionados ao processamento.

E) auxilia o Gerente de Projetos de Software com técnicas para medir o esforço necessário para o desenvolvimento de um sistema, apoiando-o, também, no levantamento dos custos, análise de qualidade e análise de produtividade.

Gabarito - APF

	09 – D	17 – B	25 – D	33 - A
	10 – A	18 – E	26 - D	34 - E
	11 – B	19 – C	27 - A	35 - C
	12 – D	20 - B	28 - B	36 - B
	13 - B	21 - D	29 - B	37 - E
6 – E	14 – E	22 – A	30 - D	38 - E
7 – B	15 –B	23 - A	31 - D	
08 – B	16 – C	24 - D	32 - B	