

Exercício de Revisão N2

Data: 07/11/2019

Turma: 1º Período

Faculdade de Tecnologia SENAI de Desenvolvimento Gerencial – FATESG

Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

Componente Curricular: Engenharia de Software

Docente: Elisabete Tie Hato

Aluno: Vinicius Araujo Lopes

Conforme conteúdo ministrado em sala de aula e através de pesquisa, responda:

1. Defina as seguintes áreas do Swebok:

a. Gerência de Configuração de Software

R: Identifica a configuração do software durante o processo de desenvolvimento, em momentos distintos para controlar modificações e manter a integridade e a autoridade da configuração em todas as partes do ciclo de vida de sistema. Responsável pela avaliação e aprovação de alterações das configurações do software.

b. Gerência da Engenharia de Software

R: Inicia e define o escopo do projeto, além de planejar, aprovar, revisar e avaliar a viabilidade de execução e a disponibilidade de recursos durante todo o processo de construção do software. Também é responsável pela elaboração e o cumprimento de prazos de cronogramas, e pelo controle do processo de execução do desenvolvimento. Mensura, avalia, direciona e redireciona cada etapa do processo.

c. Qualidade de Software

R: Aborda questões além do código fonte e aparência do sistema tais como: cultura e ética da engenharia de software, valor, custo, modelos e características da qualidade. Realiza auditorias em busca da melhoria da qualidade do software considerando os valores supra.

d. Processos da Engenharia de Software

R: Define, implementa, avalia, mensura, gerencia, altera e melhora o próprio processo de engenharia de software. A partir dos modelos e técnicas definidos inicialmente, acompanha a necessidade possíveis mudanças nos processos que, uma vez identificadas, determinam mudanças no processo de engenharia.

e. Ferramentas e Métodos

R: Ferramentas criadas para auxiliar o ciclo de vida do software tais como: editores e construtores, capazes de criar modelos, construir, compilar e interpretar, testar, analisar, depurar, realizar manutenção, realizar reengenharia ou engenharia reversa, compilar e implementar, versionar e armazenar os requisitos e documentos do software em desenvolvimento.

Métodos são as técnicas e formas de utilização das ferramentas, seguindo padrões pré-estabelecidos.

2. Para cada área acima, liste três atividades executadas em cada área e uma ferramenta utilizada em cada área.

a. Gerência de Configuração de Software

R: Atividades: Planejamento para GCS, Identificação dos itens a serem controlados e Liberação e Entrega de Software.

Ferramenta: Git, para a liberação e entrega do software.

b. Gerência da Engenharia de Software

R: Atividades: Análise de Viabilidade, Gerenciamento do Planejamento e Processo de Monitoramento.

Ferramenta: Google Gantt, para o planejamento do projeto.

c. Qualidade de Software

R: Atividades: Validação, Validação e Auditoria.

Ferramenta: Google Formulários, para obter o feedback do cliente.

d. Processos da Engenharia de Software

R: Atividades: Medição do processo e do produto, medição da qualidade e adaptação do processo.

Ferramenta: Yed, para diagramação e modelagem do processo.

e. Ferramentas e Métodos

R: Modelagem de dados.

Ferramenta: BrModelo, para diagramação de classes.

3. O que é ciclo de vida de software?

R: É um conjunto de atividades, ações, marcos e produtos de trabalho que são necessários para fazer engenharia de software com alta qualidade, que ajuda a melhorar o andamento e a garantir que os objetivos finais de cada passo sejam alcançados.

4. Dentre os modelos de ciclo de vida de software, podemos destacar os seguintes modelos:

a. Caótico d. Iterativo e Incremental

b. Cascata e. RAD – Rapid Application Development

c. Prototipação f. Espiral

Apresente duas características que melhor se encaixe com cada um dos modelos acima.

a. Caótico d. Iterativo e Incremental

R: O software é entregue em partes e melhorado a cada novo ciclo, com base no aprendizado do ciclo anterior.

b. Cascata e. RAD – Rapid Application Development

R: Reuso de componentes e criação de componentes reutilizáveis.

c. Prototipação f. Espiral

R: Desenvolvimento simultâneo e paralelo dos componentes. Muito complexo para gerenciar.

5. Escolha quatro modelos de ciclo de vida de software acima e construa o diagrama de funcionamento dos modelos escolhidos.

R: 0.o

6. O que são metodologias ágeis de desenvolvimento de software?

R: São metodologias de desenvolvimento com princípios e características que tornam a construção de software mais flexível, adaptativa, interativa e rápida.

7. Quais são as características existentes em uma metodologia ágil?

R: Contato contínuo com o cliente, reformulação de planejamento, mudança de requisitos e satisfação do cliente avaliada durante o processo de desenvolvimento.

8. Quais são os valores do manifesto ágil?

R: 1. Indivíduos e interações valem mais que processos e ferramentas;
2. Software em funcionamento vale mais que documentação abrangente;
3. Colaboração com o cliente vale mais que negociação de contratos;
4. Responder a mudanças vale mais que seguir um plano.

9. Cite quatro benefícios da metodologia ágil para o desenvolvimento de software.

R: 1. O cliente participa do processo de desenvolvimento;
2. O tempo de desenvolvimento é reduzido;
3. Não perde-se tempo construindo componentes do software que não serão utilizados pelo cliente;
4. Desenvolvedores são multifuncionais e participam do desenvolvimento de áreas diferentes do produto.

10. Liste quatro metodologias ágeis existentes no mercado:

R: FDD, DSDM, DAS e XP.

a. Apresente uma breve síntese de cada metodologia escolhida.

b. Apresente o funcionamento de cada metodologia escolhida.

R(a) FDD: Busca o desenvolvimento por funcionalidade, ou seja, por um requisito funcional do sistema.

R(b) FDD: Desenvolve-se um modelo generalizado e constrói-se uma lista de features. Para cada uma das features da lista é realizado o planejamento o projeto e a construção.

R(a) DSDM: Originalmente baseada em "Desenvolvimento Rápido de Aplicação" (RAD) possui foco no envolvimento constante do cliente em todas as fases.

R(b) DSDM: A metodologia é dividida em Pré-Projeto, Ciclo de vida do projeto e Pós-projeto. Seleciona-se um dos projetos candidatos, define-se o orçamento e o contrato. Estuda-se a viabilidade e é feito um estudo do negócio em constante contato com o cliente. Implementa-se e, após a entrega é feita uma avaliação da eficiência e eficácia do projeto com base na qualidade do produto, podendo retornar a passos anteriores para refinamento da solução.

R(a) DAS: Possui um ciclo definido por três etapas: especulação, colaboração e aprendizagem.

R(b) DAS: Especulam-se os requisitos e suas prioridades, define-se o plano de desenvolvimento. O desenvolvimento ocorre em prazos curtos pré-fixados. O cliente e a equipe de desenvolvimento, aprendem à medida em que as necessidades aparecem e suas soluções são definidas, aprendem sobre

essas necessidades e erros cometidos e volta-se a primeira etapa, com novas prioridades, recomeçando o ciclo.

R(a) XP: Cria sistemas de melhor qualidade, são produzidos em menos tempo e de forma mais econômica que o habitual. Prioriza a as partes mais importantes do Projeto, segundo o julgamento do próprio cliente.

R(b) XP: Faz-se a análise com base nas prioridades do cliente, concebe-se o design do produto e inicia-se a implementação. Testes são realizados e, caso aprovado, é feita a implantação do produto e posterior manutenção.

11. O que é qualidade de software?

R: É promover o desenvolvimento de um software de baixo custo, com processos bem planejados e definidos, resultando em um estável, que atenda aos requisitos de negócio do cliente, com uma boa experiência de uso para o o usuário, com baixos índices de manutenção e de erros de funcionamento. Praticamente utópica.

12. O que significa os termos maturidade e capacidade na área de qualidade de software?

R: Maturidade é a capacidade se repetir uma série de resultados de uma maneira previsível. Capacidade é a habilidade com que um processo alcança o resultado esperado.

13. O que é CMMI e MPS.Br e em quantos níveis é dividido cada um?

R: CMMI: Modelo de desenvolvimento com um enfoque voltado para a capacidade de maturidade de processos de software. É dividido em 5 níveis.

MPS.Br (Melhoria de Processos do Software Brasileiro), é um modelo de qualidade de processo criado em 2003 pela Softex (Associação para Promoção da Excelência do Software Brasileiro) para melhorar a capacidade de desenvolvimento de software nas empresas brasileiras. É dividido em 7 níveis.

14. O que são atributos de processo para o CMMI e MPS.Br?

R: São as qualidades e características esperadas de cada processo.

15. Quais níveis existem no CMMI, escreva uma breve síntese de cada nível.

R: Nível 1 - Inicial: os processos normalmente estão envoltos num caos decorrente da não obediência ou ainda, inexistência de padrões;

Nível 2 - Gerenciado: os projetos têm seus requisitos gerenciados neste ponto. Além disso, há o planejamento, a medição e o controle dos diferentes processos;

Nível 3 - Definido: os processos já estão claramente definidos e são compreendidos dentro da organização. Os procedimentos se encontram padronizados, além de ser preciso prever sua aplicação em diferentes projetos;

Nível 4 - Gerenciado Quantitativamente: ocorre o aumento da previsibilidade do desempenho de diferentes processos, uma vez que os mesmos já são controlados quantitativamente;

Nível 5 - Otimizado: existe uma melhoria contínua dos processos.

(Fonte: <https://www.devmedia.com.br/maturidade-no-desenvolvimento-de-software-cmmi-e-mps-br/27010>)

16. Quais níveis existem no MPS.Br, escreva uma breve síntese de cada nível.

R: A – Em Otimização: há a preocupação com questões como inovação e análise de causas;

B – Gerenciado Quantitativamente: avalia-se o desempenho dos processos, além da gerência quantitativa dos mesmos.

C – Definido: aqui ocorre o gerenciamento de riscos.

D – Largamente Definido: envolve verificação, validação, além da liberação, instalação e integração de produtos, dentre outras atividades.

E – Parcialmente Definido: considera processos como treinamento, adaptação de processos para gerência de projetos, além da preocupação com a melhoria e o controle do processo organizacional.

F – Gerenciado: introduz controles de medição, gerência de configuração, conceitos sobre aquisição e garantia da qualidade.

G – Parcialmente Gerenciado: neste ponto inicial deve-se iniciar o gerenciamento de requisitos e de projetos.

(Fonte: <https://www.devmedia.com.br/maturidade-no-desenvolvimento-de-software-cmmi-e-mps-br/27010>)

17. Apresente, no mínimo, três diferenças existentes entre o CMMI e o MPS.Br.

R: 1. CMMI é recomendável para grandes fábricas de software enquanto o MPS-Br é uma alternativa ao CMMI para fábricas de médio e pequeno porte.

2. CMMI necessita de investimento financeiro de alto nível para a implementação desse modelo de desenvolvimento, enquanto o MPS-BR requer menor volume de investimento.

3. CMMI exige mudanças drásticas de postura para a evolução entre os níveis, enquanto o MPS-BR tem mais níveis e é mais gradual, causando menor impacto nos processos internos.

18. O que é Seis Sigma? Quais suas fases?

R: É uma ferramenta para identificar e implementar melhorias nos processos internos de uma empresa, garantir custos de operação menores e, consequentemente, aumentar os lucros a partir da estratégia DMAIC (Definir, Mensurar, Analisar, Incrementar e Controlar).

Fases: formação (forming), agitação (storming), normalização (norming) e desempenho (performing).

(Fontes: <https://endeavor.org.br/estrategia-e-gestao/seis-sigma/> e <https://www.fm2s.com.br/fases-de-uma-equipe-6-sigma/>)

19. O que é Balanced Scorecard (BSC)?

R: é uma nova metodologia para medir o desempenho das organizações e, ao mesmo tempo, definir estratégias e fazer planejamentos de uma forma muito mais ampla e abrangente, sem estar engessado a métricas unicamente focadas em resultados financeiros.

(Fonte: <https://www.agendor.com.br/blog/balanced-scorecard-conceito/>)

20. Pesquisar sobre “Processo Unificado da Rational” e responder: O que é RUP?

Descreva suas fases e disciplinas.

R: é um processo proprietário de Engenharia de software, criado para apoiar o desenvolvimento orientado a objetos, fornecendo uma forma sistemática para se obter vantagens no uso da UML. Criado pela Rational Software Corporation, adquirido pela IBM, ganhando um novo nome IRUP que agora é uma abreviação de IBM Rational Unified Process com o objetivo de aumentar a sua produtividade no processo de desenvolvimento.

Fases: Concepção, Elaboração, Construção e Transição.

Disciplinas de Engenharia: Modelagem de Negócios, Requisitos, Análise e Projeto, Implementação, Teste e Implantação.

Disciplinas de Apoio/Suporte: Ambiente, Configuração e Gerência de Mudança, e Gerência de Projeto.

(Fontes: <https://www.infoescola.com/engenharia-de-software/rup/>
e https://pt.wikipedia.org/wiki/IBM_Rational_Unified_Process)

essa última merece só ½ certo... ;)
