



Fundação CECIERJ – Vice-Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Engenharia de Software

AP1 2º semestre de 2016.

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
 2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
 3. Você pode usar lápis para responder as questões.
 4. Ao final da prova devolva as folhas de questões e as de respostas.
 5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.
-

1. Explique porque os modelos são importantes e cite exemplos de como eles ajudam no desenvolvimento de software. (2,0 pontos).

Modelos são úteis para o entendimento de problemas complexos, em que todos os detalhes não podem ser mantidos em mente ao mesmo tempo, para a difusão do conhecimento entre os componentes da equipe de projeto e testar ideias e hipóteses antes de realizá-las, sendo mais barato ver o resultado destas ideias em um modelo do que construir um elemento (ex. software) real para validá-las.

No desenvolvimento de software, modelos revelam as características essenciais de um projeto, filtrando detalhes irrelevantes que dificultam o entendimento do sistema. Este é o princípio da abstração, que facilita a compreensão do sistema, a comunicação entre os integrantes da equipe e serve como linguagem comum para expressar diferentes versões do sistema.

2. Pergunta da AD1: Explique o que são requisitos e cite que tipos de requisito devem ser descritos para apoiar a construção de um software. (2,0 pontos).

Requisitos são declarações do que um sistema deve fazer, que dependem das tarefas cuja execução será suportada pelo sistema e das necessidades específicas da

organização que o utilizará. São características ou funções que os sistemas devem possuir para que atendam aos objetivos desta organização e resolvam um conjunto de problemas. Os principais tipos de requisitos são funcionais (descrições das funções que o sistema deve prover para o usuário, ligados ao domínio do problema) e não-funcionais (restrições impostas sobre as funções oferecidas pelo sistema, ligadas ao domínio da solução). A principal atividade ligada aos requisitos é a elicitación (ou análise) de requisitos. Ela é a primeira atividade do ciclo de vida em cascata, precedendo o projeto (*design*) do software. Nos ciclos incrementais, ela também é a primeira atividade, mas é executada diversas vezes, seguindo os diversos incrementos do ciclo de vida. Outra atividade intimamente relacionada aos requisitos é a atividade de testes, que usa a descrição dos requisitos como referência para validar a implementação do software.

3. Quais são os três principais componentes de um diagrama de casos de uso? O que cada um destes componentes representa no diagrama? (2,0 pontos).

Atores, que representam elementos externos ao sistema, mas que interagem com ele; casos de uso, que são um conjunto de ações que um ator pode realizar através do sistema e que lhe oferece algum benefício; e relacionamentos, que podem ocorrer entre atores e casos de uso (participação), entre atores (generalização) e entre casos de uso (extensão e inclusão).

4. Explique a diferença entre um modelo de ciclo de vida incremental evolutivo e um ciclo de vida incremental descartável. (2,0 pontos).

Em um modelo de ciclo de vida incremental evolutivo, os resultados produzidos em uma fase do ciclo de desenvolvimento são mantidos para o próximo incremento, sendo evoluídos e complementados ao longo do tempo. Nos ciclos descartáveis, os resultados produzidos em um incremento são descartados ao final deste incremento, sendo construído do zero no próximo incremento (em tese, com maior qualidade).

5. Qual é a ordem correta das atividades em um ciclo de vida em cascata? (1,0 ponto).
- Projeto, Requisitos, Codificação, Testes, Manutenção
 - Requisitos, Projeto, Manutenção, Codificação, Testes
 - Requisitos, Projeto, Codificação, Testes, Manutenção
 - Requisitos, Codificação, Projeto, Testes, Manutenção
 - Requisitos, Codificação, Testes, Projeto, Manutenção

Opção C.

6. Qual dos modelos abaixo não é parte da modelagem de sistemas orientada a objetos com UML? (1,0 ponto)
- a. Diagrama de classes
 - b. Diagrama entidade-relacionamento
 - c. Diagrama de casos de uso
 - d. Diagrama de colaboração
 - e. Diagrama de estados

Opção B.

Boa Prova!