



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina de Arquitetura e Projeto de Sistemas II

Gabarito - AP1 2º semestre de 2009.

Nome –

Assinatura –

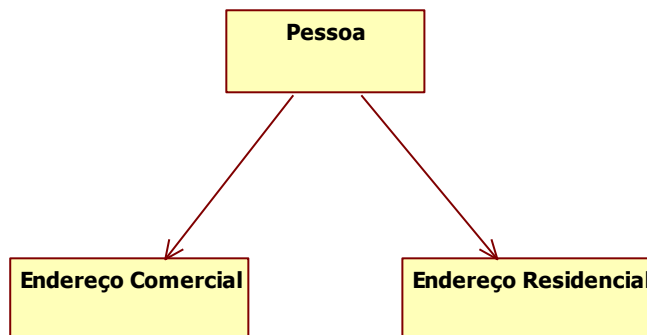
Observações:

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.
-

Questão 1 (2 pontos)

Analise o diagrama de classes a seguir e responda,

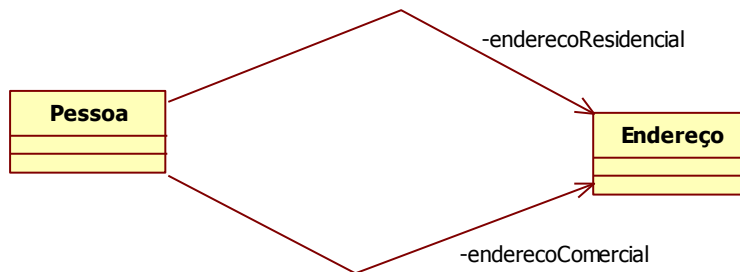
- (a) Qual heurística de projeto está sendo ferida por esse diagrama? Justifique a sua resposta.
- (b) Redesenhe o diagrama de forma que ele não fira mais nenhuma heurística de projetos.



Resposta:

- (a) A heurística que está sendo ferida é: “Tenha certeza que as abstrações são realmente classes, e não simplesmente papéis”. As classes Endereço Comercial e Endereço Residencial na verdade representam a mesma abstração: Endereço. O fato de ser comercial ou residencial se refere ao papel que a classe Endereço exerce em cada relacionamento.

(b)



Questão 2 (3 pontos)

Para o diagrama de classes conceitual a seguir,

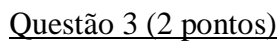
- (a) Determine o grau de dependência direto e indireto para a classe “ItemPedido” em relação às demais classes do diagrama, justificando a resposta via listagem das classes envolvidas em cada uma das contagens de grau de dependência (Lembrete: a navegabilidade nos casos de todo-parte é sempre no sentido do todo para a parte. Ou seja, em situações de composição ou agregação, a parte não conhece o todo. Além disso, a inexistência de setas nas associações representa navegabilidade bidirecional.).

Resposta: Grau de dependência direto para a classe ItemPedido: 1 (classe Produto). Grau de dependência indireto para a classe ItemPedido: 3 (classes Produto, Especificação e ItemEstoque).

- (b) Assumindo que o espaço-estado do atributo “quantia” da classe “Pagamento” seja de R\$ 50,00 a R\$ 500,00, é permitido que a classe “PagamentoCheque” modifique esse espaço-estado para de R\$ 0 a R\$ 1000,00? Justifique a sua resposta.

Resposta: Não. Tendo em vista que todo PagamentoCheque é um Pagamento, e que a variação de espaço-estado em casos de herança deve ser mais restritiva nas subclasses, a modificação proposta é inválida. Ou seja, o intervalo entre 0 e 1000 não está dentro do intervalo entre 50 e 500.

Resposta: O mecanismo de invariante de classe serve para garantir que as restrições de espaço-estado sejam respeitadas. As invariantes de classe devem ser válidas para todos os objetos em equilíbrio (sem método em execução) da classe. As invariantes da classe Pagamento são: (1) existir um pedido associado e (2) o atributo quantia ser equivalente ao custo total do pedido (quantidade de ItemPedido multiplicado pelo preço de cada Produto).



(a) Descreva suas principais características e dificuldades.

Resposta: Trata-se de um desenvolvimento que envolve um processo iterativo, evolutivo e ágil que explora a reutilização de software e oferece uma representação conceitual única durante o processo de desenvolvimento. Algumas de suas dificuldades são: a

identificação de casos de uso, a classificação e organização de classes e objetos e ainda o detalhamento do comportamento de classes e objetos.

(a) Qual é o princípio básico da orientação a objetos e qual é o seu objetivo?

Resposta: O princípio básico é o encapsulamento, cujo objetivo é restringir o escopo de informação para atingir legibilidade, manutenibilidade e reutilizabilidade. O encapsulamento proíbe a visualização da realização de uma entidade.

Questão 4 (3 pontos)

Em relação à Engenharia de Requisitos,

(a) Quais são as suas dificuldades (descreva ao menos 3)?

Resposta: Dificuldades: (i) Comunicação com usuário: problemas de comunicação, ocasionados por ruídos no canal (meio) ou mesmo pela falta de compreensão do código utilizado (linguagem), são significativos para o entendimento dos requisitos a serem contemplados pelo projeto do sistema, o que pode resultar na agregação de funcionalidades erradas ou mesmo desnecessárias ao produto final; (ii) Mudanças constantes: solicitações realizadas pelos clientes são passíveis de alterações frequentes, devido a mudanças gerenciais, de lógica de negócio, do arcabouço tecnológico, dentre outras, e isso requer uma postura flexível e dinâmica dos engenheiros de software; e (iii) Diferentes formas de representação: a falta de padronização de uso de uma forma para se representar o problema a ser resolvido e os requisitos a serem contemplados pelo sistema podem gerar distorções entre desenvolvedores e gerentes e entre gerentes e clientes, ocasionando tensões desnecessárias, por falta de planejamento.

(b) Discorra sobre três técnicas de elicitação de requisitos.

Resposta: Técnicas: (i) Entrevista: é uma técnica simples, muito conhecida e usada, mas que deve ser conduzida de forma estruturada a fim de evitar se tornar muito informal – menos produtiva – e consistir apenas em uma conversa; (ii) Reuniões de Brainstorming: consistem em reuniões de trocas de idéias onde várias questões são levantadas e colocadas em pauta, começando do nível mais geral e prosseguindo-se para um detalhamento mais específico de idéias sobre a construção do sistema; e (iii) Prototipação: é uma técnica prática e bastante utilizada, na qual um entendimento inicial do sistema pode permitir a construção de um pequeno protótipo do sistema, a fim de exibir ao cliente o que foi assimilado *a priori* pelo engenheiro de requisitos acerca do que vai ser o sistema.