



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 da AP1 – 1º semestre de 2013

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)

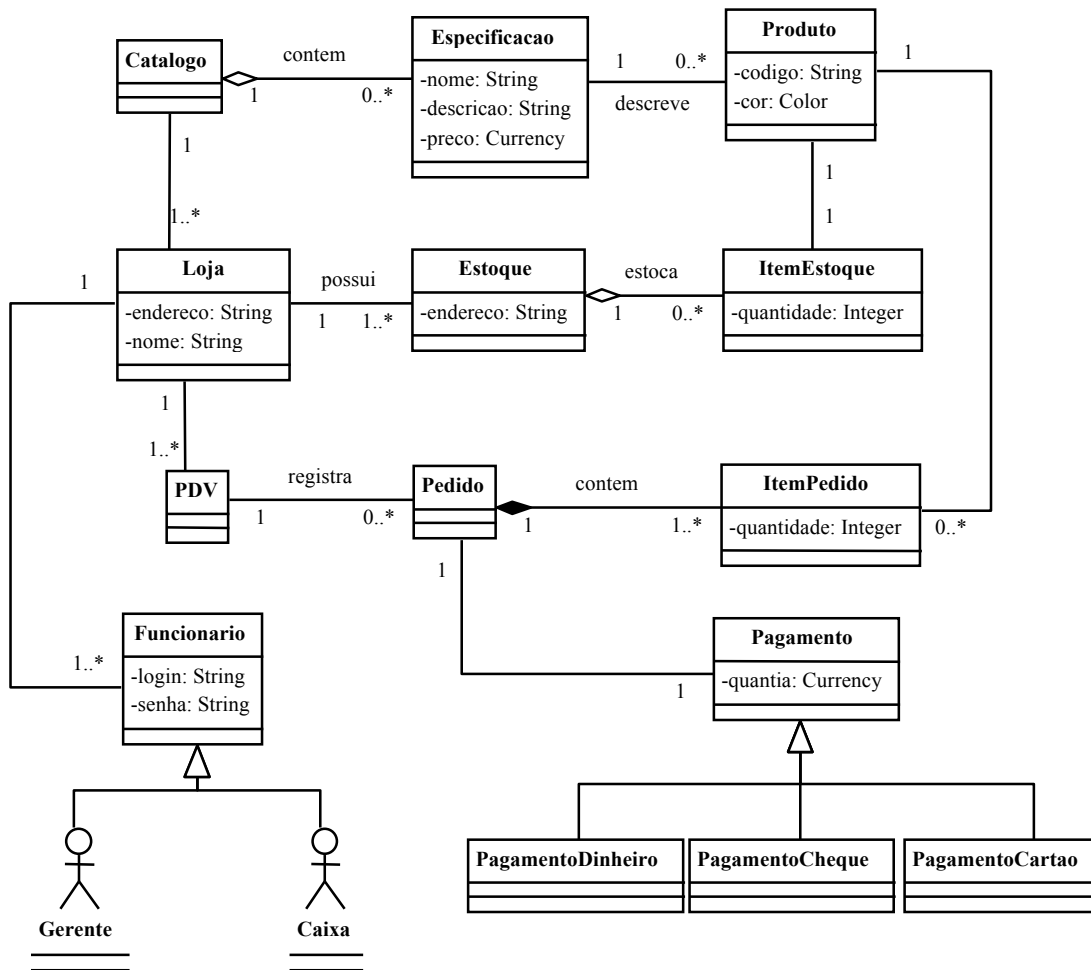
Relacione a coluna da esquerda com a coluna da direita.

- | | |
|--------------------------------------|-----------------------------------------------------------------------------------------------------------|
| (a) diagrama de classes | (1) Detalha quais classes são mais prioritárias no sistema. |
| (b) diagrama de casos de uso | (2) Exibe a estrutura da interface com o usuário. |
| (c) diagrama de transição de estados | (3) Detalha o comportamento de um objeto no decorrer da sua vida. |
| (d) diagrama de sequência | (4) Descreve um determinado estado de execução do sistema em termos dos objetos. |
| (e) descrição de casos de uso | (5) Explicita a estrutura estática interna do sistema. |
| | (6) Explicita as possibilidades de interação entre os usuários e o sistema. |
| | (7) Detalha a interação entre diferentes objetos do sistema para atender a uma funcionalidade específica. |
| | (8) Detalha uma determinada possibilidade de interação entre o usuário e o sistema. |

Resposta: a → 5; b → 6; c → 3; d → 7; e → 8

Questão 2 (3 pontos)

Para o diagrama de classes conceitual a seguir, responda com suas palavras:



- (a) Determine o grau de dependência direto e indireto para a classe “PagamentoCheque” 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: nas associações 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 PagamentoCheque: 1 (classe Pagamento). Grau de dependência indireto para a classe PagamentoCheque: 11 (classes Pagamento, Pedido, ItemPedido, Produto, Especificação, ItemEstoque, PDV, Loja, Funcário, Estoque e Catalogo).

- (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\$ 300,00 a R\$ 500,00? Justifique a sua resposta.

Resposta: Sim. Tendo em vista que 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 válida. Ou seja, o intervalo de 300 a 500 está contido no intervalo de 50 a 500.

- (c) A qual domínio (base, arquitetura, negócio ou aplicação) pertence a classe “Loja”? Justifique a sua resposta.

Resposta: A classe Loja pertence ao domínio de negócio. Por se tratar de um diagrama de classes conceitual, com o propósito de explicitar as entidades de negócio, todas as classes pertencem ao domínio de negócio, representando entidades do espaço do problema que está sendo tratado.

Questão 3 (2 pontos)

Indique se a afirmação é verdadeira ou falsa:

- (a) Modelos ajudam a visualizar, entender e construir sistemas complexos de software pois capturam as partes essenciais de um sistema de software, abstraindo-se dos detalhes de implementação.
- (b) UML não contempla e especificação de Casos de Uso como forma de Representar Requisitos Funcionais.
- (c) Polimorfismo é uma técnica que não fomenta a implementação de um sistema de software de fácil adaptação.
- (d) As atividades do Processo de Elicitação de Requisitos são: elicitação, análise e negociação, especificação, modelagem, validação e gerência
- (e) Casos de Uso não abstraem um funcionalidade do Sistema uma vez são usados para representar o processo do sistema.

Resposta: a -> V; b->F; c-> F; d-> V; e-> F

Questão 4 (3 pontos)

A construção de sistemas de software se baseia em na aplicação de 3 princípios: Abstração, Encapsulamento e Decomposição. Explique estes princípios, identificando seus benefícios e exemplifique com um cenário ilustrativo (1 ponto para cada princípio).

Resposta:

Abstração: Princípio que permite o desenvolvedor concentrar nas partes essenciais de um projeto, filtrando detalhes irrelevantes que dificultam o entendimento do sistema.

Ex: Ao se projetar um Carro em um Sistema de Estacionamento, a informação Potência do Motor pode ser abstraída uma vez que tal informação não é importante para o entendimento do sistema.

Encapsulamento: Princípio que restringe o acesso a partes de um módulo (ex: classe) de modo que elementos externos não dependam da parte restrita. Tal princípio facilita a manutenção de sistemas.

Ex: Ao se projetar uma classe Lista, a forma de armazenamento dos elementos da lista devem ser escondida (encapsulada).

Decomposição : Princípio que permite quebrar um problema complexo e difícil de ser tratado, em subproblemas menores e de solução mais “simples”.

Ex: Ao se projetar a funcionalidade de um Sistema de Estacionamento, esta deve ser decomposta em vários Casos de Uso de modo que cada caso possa ser tratado isoladamente. Após a definição e implementação dos casos de uso, estes devem ser integrados para representar o sistema em desenvolvimento.