



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 AP3 – 2º semestre de 2012

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.
-

Relacione cada elemento da coluna da esquerda com um e somente um elemento na coluna da direita.

- | | |
|--------------------------------------|---|
| (a) diagrama de classes | (1) Detalha o comportamento de um objeto no decorrer da sua vida. |
| (b) diagrama de casos de uso | (2) Explicita a estrutura estática interna do sistema. |
| (c) diagrama de transição de estados | (3) Explicita as possibilidades de interação entre os usuários e o sistema. |
| (d) diagrama de sequência | (4) Explicita a eficiência do sistema em termos de utilização de recursos. |
| (e) descrição de casos de uso | (5) Detalha a implantação do sistema em ambiente de uso. |
| | (6) Detalha a interação entre diferentes objetos do sistema para atender a uma funcionalidade específica. |
| | (7) Detalha uma determinada possibilidade de interação entre o usuário e o sistema. |
| | (8) Explicita como o estado dos objetos é persistido em banco de dados. |

Resposta: a → 2; b → 3; c → 1; d → 6; e → 7

Questão 2 (3 pontos)

Para cada uma das afirmativas a seguir, indique se ela é correta ou incorreta e justifique a resposta com as suas palavras.

- (a) “Em um sistema com interface com o usuário, as classes de modelo devem depender das classes de interface”.

Resposta: Incorreta. Para facilitar a reutilização das classes de modelo e eventual substituição da interface, elas não devem conter dependência para nenhuma classe de interface.

- (b) “Minimize o número de mensagens enviadas entre uma classe e as suas classes colaboradoras”.

Resposta: Correta. Um número excessivo de mensagens entre classes pode indicar que o encapsulamento das classes está sendo quebrado, e que métodos com mais responsabilidades são necessários.

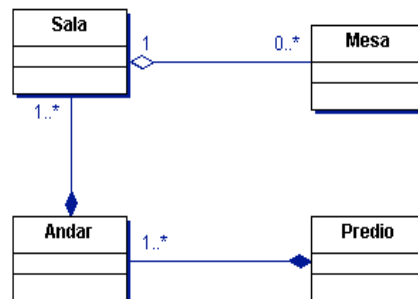
- (c) “Para aumentar a encapsulamento do sistema, sempre faça uso de atributos protegidos”.

Resposta: Incorreta. Para que o encapsulamento seja aumentado, é necessário fazer uso de atributos privados, e fornecer métodos de acesso a esses atributos.

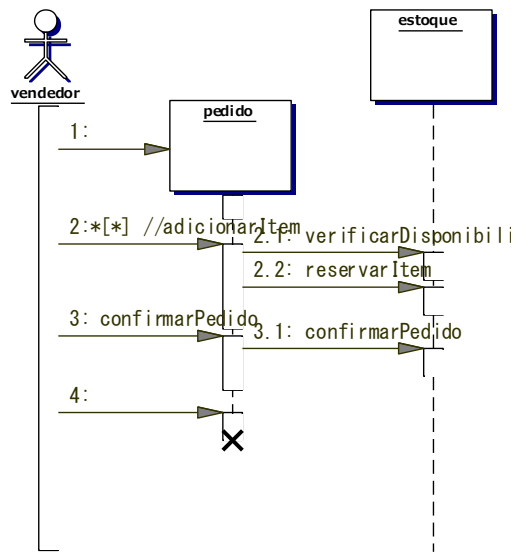
Questão 3 (2 pontos)

Responda com Verdadeiro (V) ou Falso (F) as afirmações abaixo:

- a) () Modelagem é uma técnica utilizada para a disseminação de conhecimento entre os participantes do processo de desenvolvimento de um sistema de software.
- b) () Na especialização de Frameworks Caixa-Preta o framework recebe um conjunto de parâmetros que representa o comportamento específico da aplicação.
- c) () No diagrama abaixo, o Ator Mesa interage com o Ator Sala



- d) () Em Orientação a Objetos, Herança é um conceito que fomenta reutilização através do reaproveitamento das definições da superclasse pela subclasse.
- e) () O Diagrama abaixo não há indícios que a classe *Estoque* possui um método *reservarItem()*.



Resposta: a → V; b → V; c → F; d → V; e → F

Questão 4 (3 pontos)

Dado os requisitos abaixo especifique:

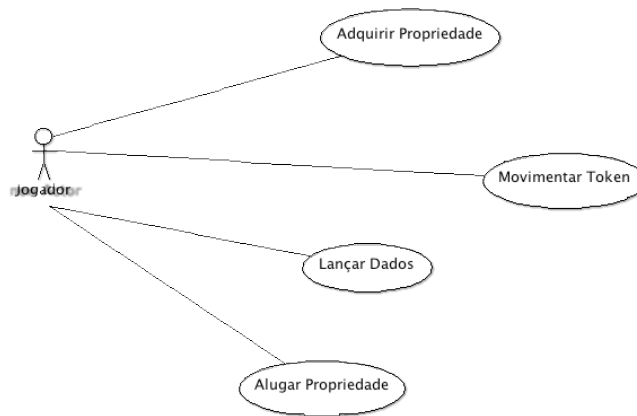
- (1 Ponto) Diagrama de Caso de uso com no máximo dois atores e 4 casos de uso.
- (1 Ponto) Diagrama de Classes com no máximo 6 classes.
- (1 Ponto) Mostre como se dá o rastreamento entre os termos presentes nos requisitos e os elementos de modelagem na forma:
Classe XXXX presente no Requisito YYYY

Sua empresa foi contratada para desenvolver um jogo de estratégia equivalente ao Monopólio. Para tal você recebeu um *Documento de Visão* que apresenta minimamente os requisitos do sistema. Um extrato deste documento é apresentado a seguir:

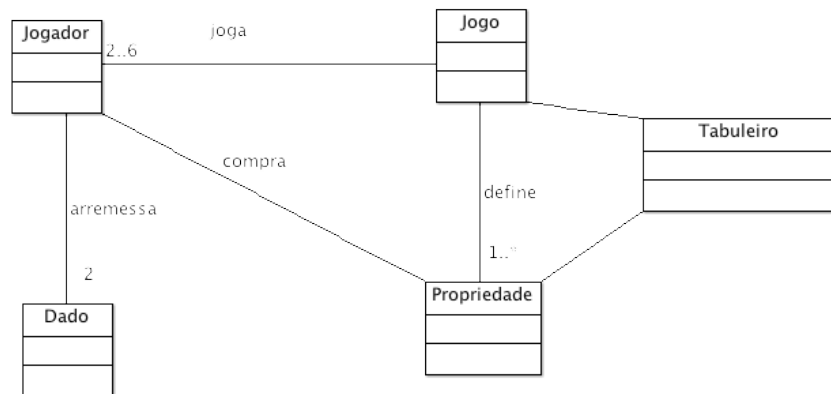
- O Jogo *MonoP* tem como objetivo simular o jogo de Monopólio, com as mesmas funcionalidades presentes no jogo de tabuleiro.
- O Jogo permite a aquisição e aluguel de propriedades dispostas em um mapa.
- A navegação entre as propriedades deverá ser sequencial e unidirecional.
- O máximo de jogadores permitidos no jogo será 6 e o mínimo 2.
- Cada jogador lança um conjunto com dois dados, cujos valores serão somados. Esta soma indicará a movimentação do jogador no mapa de acordo com a fórmula : Posição Futura = Posição Atual + Soma dos Dados.

6. Após a movimentação, quando um Jogador chega em uma propriedade, ele pode comprá-la, caso não tenha dono. Caso a propriedade tenha um dono (diferente do próprio jogador), um aluguel será cobrado.
7. Cada Jogador é identificado por uma cor no início do jogo.
8. Cada Jogador receberá uma Quantia em Dinheiro do Banco no início do Jogo.
9. Quando um Jogador adquire uma propriedade, ele deverá adquirir casas (até 4) para serem dispostas na propriedade corrente. Quanto mais casas, mais aluguel poderá ser cobrado a outros jogadores.
10. O Jogador só adquire uma propriedade caso o valor em dinheiro que ele possuir for maior que o valor a ser gasto.
11. No processo de compra o Jogador entrega ao Banco o valor em dinheiro da propriedade e recebe deste banco uma carta com o “Título de Propriedade”.

Resposta
a-



b-



C-

Ator Jogador – Presente em Requisitos.4 => O máximo de jogadores permitidos...

Caso de Uso Adquirir Propriedade – Presente em Requisitos.2 => O Jogo permite a aquisição e aluguel de propriedades

Caso de Uso Lançar Dados – Presente em Requisitos.5 => Cada jogador lança um conjunto com dois dados

Classe Tabuleiro - Presente em Requisitos.2 =>propriedades dispostas em um mapa.

...

Classe Propriedade - Presente em Requisitos. 2 => O Jogo permite a aquisição e aluguel de propriedades

Classe Dado - Presente em Requisitos. .5 => Cada jogador lança um conjunto com dois dados