



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina de Arquitetura e Projeto de Sistemas
Gabarito da AP1 – Polo Rocinha – 1º semestre de 2018.

Nome –

Assinatura –

Observações:

1. Prova sem consulta e sem uso de dispositivos (e.g., celular, calculadora, etc.).
 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)

Em relação ao diagrama de classes UML,

- (a) Explique com suas palavras a diferença entre os tipos de relacionamento entre classes (associação, agregação, composição e generalização)?

Resposta: Associação simples indica que duas classes colaboram, ambas com o mesmo nível de importância. Agregação é uma associação onde uma das partes exerce o papel de “todo” e a outra parte exerce o papel de “parte”. Composição é uma variação de agregação que introduz mais força na relação todo-parte, onde a “parte” pertence a somente um “todo” e tem o seu ciclo de vida atrelado a ele. Generalização é um relacionamento entre tipos que pertencem a uma mesma hierarquia de herança, onde um é mais genérico e outro é mais específico.

- (b) Cite três dos modificadores de visibilidade de métodos existentes e explique com suas palavras para que servem?

Resposta: Privado, onde somente métodos da própria classe podem acessar o método em questão; Protegido, onde métodos de outras classes que herdam da classe em questão podem acessar o método em questão; e Público, onde qualquer método de qualquer classe pode acessar o método em questão.

Questão 2 (3 pontos)

Analise o texto e responda aos itens a seguir.

“O Banco XPTO criou um novo tipo de investimento (II), onde qualquer cliente pode aplicar de R\$ 100 a R\$ 5000 e, no mesmo instante, fica sabendo qual será a sua taxa de juros para o valor investido. A taxa de juros varia de 0,5% a 0,8% ao mês em função do valor investido. O cálculo exato da taxa de juros é feito por uma classe fornecida pelo Banco Central, denominada ‘Calculador de Juros’. Contudo, somente clientes sem registro de calotes no SERASA podem fazer uso desse novo tipo de investimento. O SERASA também fornece uma classe denominada ‘Verificador de Calotes’ que permite essa análise”

- (a) Defina o contrato (invariantes, pré-condições e pós-condições) para o método responsável por efetuar o investimento de um dado valor em nome de um dado cliente. Esse método retorna a taxa de juros obtida ou uma exceção no caso do cliente estar com registro de calote no SERASA.

Resposta: Assumindo a assinatura *investe(valor : Moeda, cliente : Cliente) : real* para o método em questão, onde o retorno do método representa a taxa de juros obtida, é possível definir:

Invariantes:

Cliente sem registro de calote no SERASA

Pré-condições:

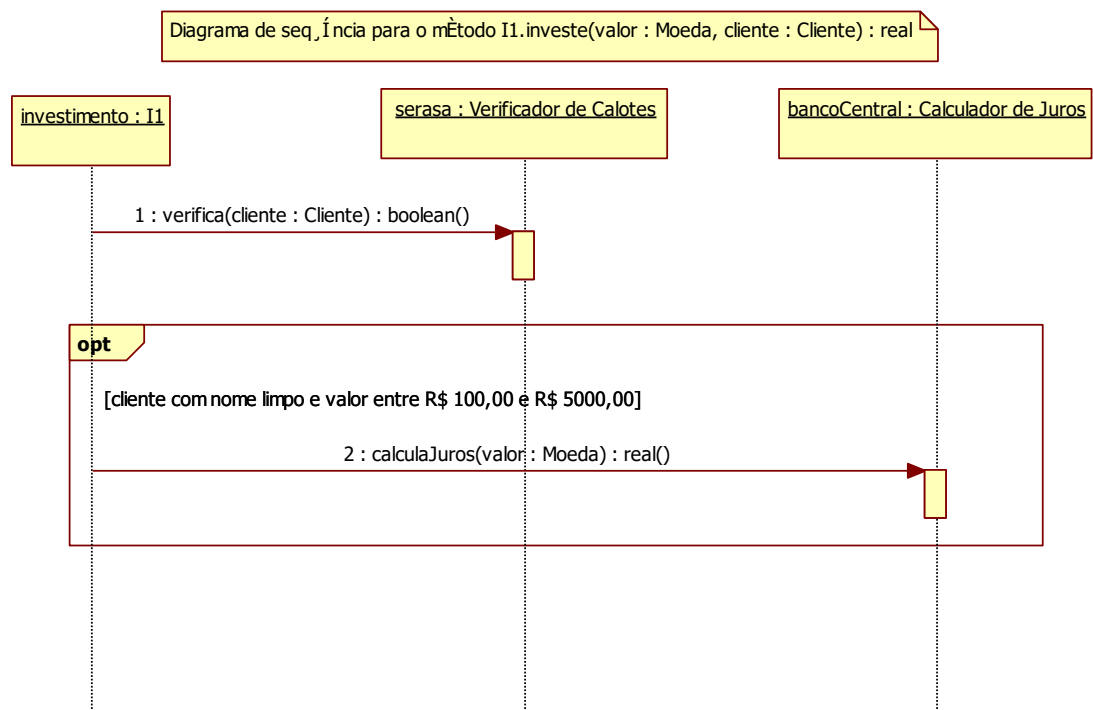
$100 \leq \text{valor} \leq 5000$

Pós-condições:

$0,5\% \leq \text{taxa de juros} \leq 0,8\%$

- (b) Faça o diagrama de sequência referente a esse método.

Resposta:



- (c) Caso esse método fosse reimplementado por uma subclasse que pretende criar o investimento I2, qual espaço-estado seria aceitável para o valor a ser aplicado e para a taxa de juro sem que o contrato original da classe referente ao investimento I1 fosse quebrado?

Resposta: Para que o investimento I2 fosse compatível com o investimento I1, seria necessário que o espaço-estado do argumento valor da classe I2 fosse igual ou menos restritivo que da classe I1 (contravariação) e o espaço-estado da taxa de juros da classe I2 fosse igual ou mais restritivo que da classe I1 (covariação). Ou seja, o espaço estado do argumento valor teria que conter o espaço-estado atual (de 100 a 5000) e o espaço estado da taxa de juros teria que estar contido no espaço-estado atual (de 0,5% a 0,8%). Por exemplo, um espaço-estado aceitável para o argumento valor seria de 50 a 10000, e um espaço-estado aceitável para a taxa de juros seria de 0,6% a 0,8%.

Questão 3 (5 pontos)

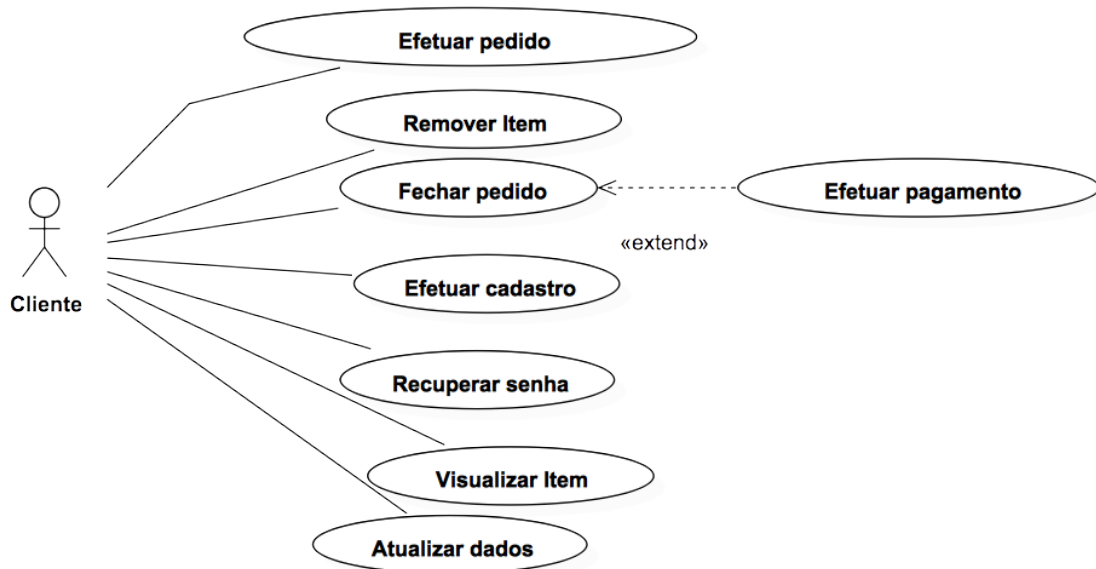
Considere a listagem abaixo de requisitos funcionais de um módulo de controle de pedidos de um supermercado *on-line*, obtida a partir de entrevistas de levantamento.

RF1:	O sistema deve permitir ao cliente efetuar pedidos compostos de itens identificados por nome e com um valor.
RF2:	O sistema deve permitir ao cliente remover itens de pedidos antes de fechá-los.
RF3:	O sistema deve permitir ao cliente fechar pedidos.

RF4:	O sistema deve permitir ao cliente efetuar pagamento de pedido fechado.
RF5:	O sistema deve permitir ao cliente atualizar seus dados cadastrais.
RF6:	O sistema deve permitir ao cliente recuperar sua senha.
RF7:	O sistema não deve permitir acesso de nenhum usuário às senhas cadastradas.
RF8:	O sistema deve permitir o cadastro de clientes com CPF, nome, endereço, usuário e senha.

(a) (1 ponto) Elabore o diagrama de casos de uso do módulo.

Resposta:



(b) (2 pontos) Apresente a descrição detalhada para o caso de uso em que o cliente fecha um pedido e efetua o pagamento com base nos requisitos levantados. Utilize o *template* de descrição a seguir.

Nome:	<definir o nome do caso de uso>
Objetivo:	<descrever o objetivo do caso de uso>
Requisitos:	<identificação dos requisitos sendo atendidos pelo caso de uso>
Atores:	<listar os atores que interagem com o caso de uso>
Trigger:	<definir que evento dispara a execução desse caso de uso>
Fluxo Principal:	<descrever passos numerados do fluxo principal do caso de uso>
Fluxo Alternativo:	<descrever os passos dos fluxos alternativos do caso de uso, indicando que evento dispara cada um deles. Cada fluxo deve ser nomeado <Número do fluxo principal>.<Número do fluxo alternativo>. Exemplo: 3.1, 3.2, 4.1.

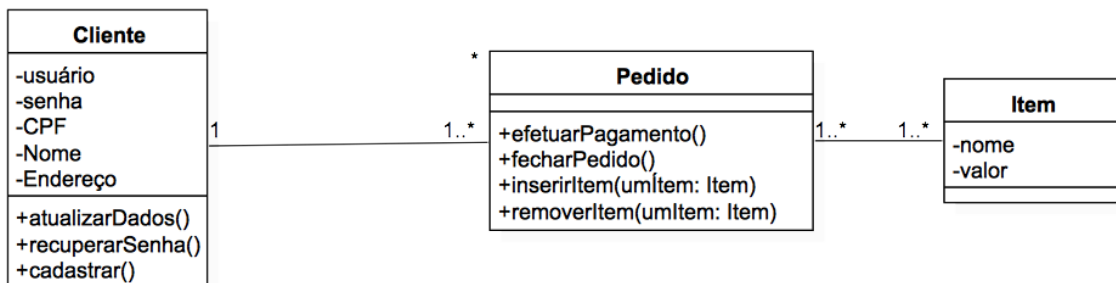
Regras de negócio:	<listar as regras de negócios que devem ser respeitadas na execução do caso de uso. Cada regra deve ser nomeada RN1, RN2 etc., e ser referenciada em passos do fluxo principal e/ou alternativo.
---------------------------	--

Resposta:

Nome:	Fechar pedido e efetuar pagamento
Objetivo:	Permite ao cliente fechar um pedido efetuado e realizar o pagamento, posteriormente
Requisitos:	RF3, RF4
Atores:	[A2] Cliente
Trigger:	Um pedido foi efetuado por um cliente [A1]
Fluxo Principal:	<ol style="list-style-type: none"> 1. Um pedido foi efetuado [A1] 2. O cliente fecha o pedido [A1][RN1] 3. O cliente efetua o pagamento [A1] 4. O sistema valida a requisição
Fluxo Alternativo:	<ol style="list-style-type: none"> 1. a) O cliente remove um item antes de fechar o pedido 1. O cliente remove um item [A1]
Regras de negócio:	[RN1] O pedido deve ser fechado para posteriormente efetuar o pagamento

(c) (1 ponto) Elabore o diagrama de classes do módulo.

Resposta (métodos “get” foram omitidos):



(d) (1 ponto) Apresente o diagrama de sequência para a sequência de quando um cliente efetua um pedido e depois remove um item.

Resposta:

interactionDiagrama de sequência

