

Análise e Projeto de Software

Atividade Avaliativa do tema Padrões de Projeto

Responder as questões abaixo da seguinte forma:

a) um relatório feito em editor de texto contendo, para cada questão:

- modelos UML com a descrição da solução
- a explicação dos principais trechos de código-fonte
- evidência de execução (com duas telas pelo menos)

E

b) código fonte de todas as soluções (como arquivos texto)

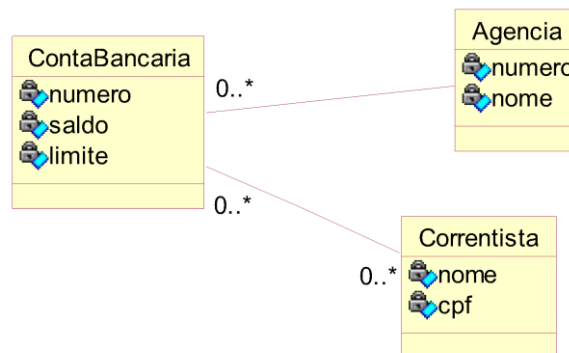
Obs: quando a solução GoF não for seguida pelo aluno, deve haver justificativa.

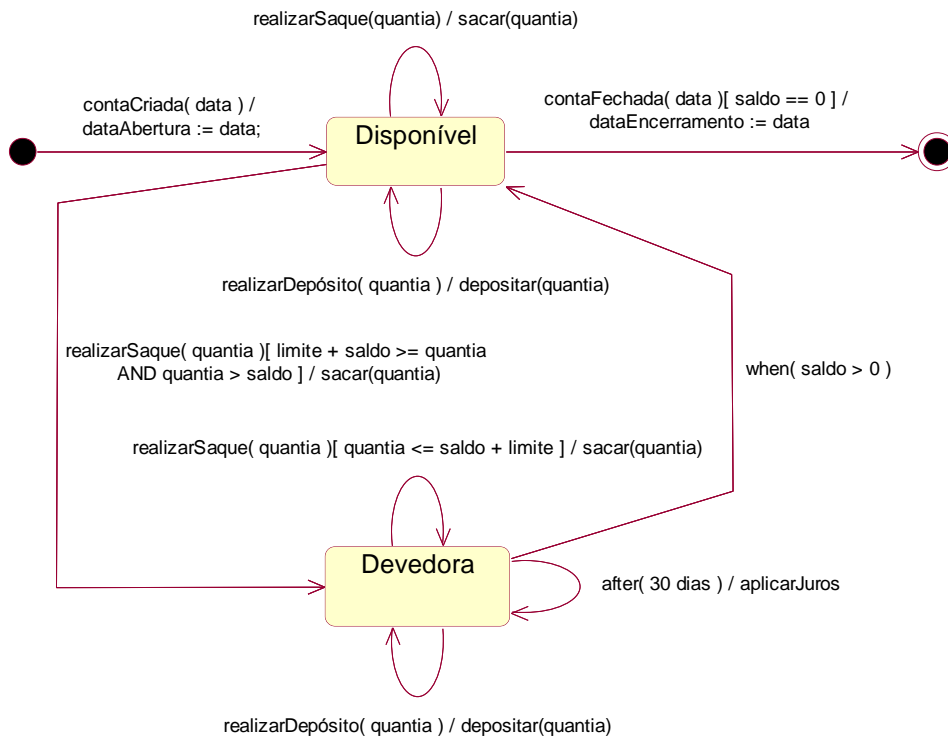
Entrega: 14/12/2016

1ª questão

Considerar o pattern State e os modelos UML a seguir.

- Desenvolver:
 - Um modelo de classes considerando a adoção do pattern State para atender os estados modelados
 - Incluir métodos e atributos adicionais definidos a partir do DTE
 - Implementar código na linguagem OO favorita
 - Fazer código que teste a troca dos estados e faça I/O simples na tela com os resultados.





2ª questão

Uma sorveteria precisa criar um sistema para cadastrar os pedidos dos seus clientes. Os sorvetes podem ser servidos em Copo (R\$0.20), Taça (sem custo) ou Casquinha (R\$1.50). Atualmente, eles contam com 5 sabores de sorvete: “Chocolate”, “Morango”, “Flocos”, “Pavê” e “Napolitano” (R\$1.50 cada bola) ou “Chocolate Diet” (R\$ 2.00 a bola). Além disso, os clientes podem acrescentar: Cobertura de Chocolate, Morango ou Caramelo (que custa R\$ 0.50).

Utilizando o Padrão Decorator crie as classes necessárias para este sistema. Apresente o modelo de classes na notação UML e a implementação do software que faça alguma I/O simples com exibição dos resultados.

3ª questão

Um texto é constituído por palavras. Quando o texto é apresentado, através do método **render**, cada palavra pode aparecer sem qualquer modificação de aspecto (utiliza-se o método **render** correspondente). É ainda possível modificar dinamicamente o aspecto das palavras, permitindo que sejam apresentadas em **negrito**, *itálico*, sublinhado, ou em combinações variadas (e.g. **negrito e itálico** ou *itálico sublinhado*, etc.). No entanto, a apresentação é sempre realizada da mesma forma (sempre através do método **render**).

Além de apresentável graficamente, um texto pode ser convertido numa cadeia de caracteres, contendo a sua informação textual (**String**). Esta operação é realizada através do método **text** (invocado sobre cada um dos elementos designados acima).

Exercício 1 – Qual o padrão de projeto é mais adequado para ser usado no desenvolvimento deste sistema?

Exercício 2 - Implemente as classes que permitem representar o texto completo, as palavras, respectivas modificações gráficas. Implemente ainda uma aplicação que ilustre o comportamento.

Represente as características gráficas da seguinte forma:

- normal `normal`
- **negrito** `negrito`
- *itálico* `<i>itálico</i>`
- sublinhado `<u>sublinhado</u>`

4a questão

Assignment 1 do arquivo

http://se.inf.ethz.ch/old/teaching/2009-S/0050/assignments/pattern_exercise.pdf

5a questão

Assignment 2 do arquivo

http://se.inf.ethz.ch/old/teaching/2009-S/0050/assignments/pattern_exercise.pdf