

Lista 3

Instruções Gerais:

- Os exercícios devem ser apresentados na mesma ordem dos enunciados e devem conter uma sequência lógica. Todos os exercícios são referentes ao mesmo sistema, portanto deve haver coerência entre eles. Os exercícios que não estiverem numa sequência lógica serão devidamente descontados.
- A Lista pode ser realizada individualmente, em dupla ou em trio, mas apenas um aluno precisa entregar a Lista por e-mail. Favor copiar os outros integrantes do grupo em cada e-mail enviado. Exceções devem ser tratadas com o próprio professor com antecedência.
- Os slides dos capítulos 5, 8 e 11 podem apoiar a realização da maioria dos exercícios desta Lista.
- Os diagramas devem ser construídos em alguma ferramenta CASE, mas a Lista deve ser entregue no formato digital no e-mail pwvendramel@gmail.com em um único arquivo PDF até às 23h00 de 12/11/2016.
- Para cada exercício em branco, incompleto ou que não atenda o enunciado, será subtraído 1 ponto da Nota de Listas conforme explicado no primeiro dia de aula. Em determinados casos, o desconto pode ser de 0,5 ponto.
- Listas com respostas suspeitas de plágio serão devidamente anuladas e “zeradas”. Os exercícios com respostas iguais entre grupos diferentes serão anulados e descontados. O aluno poderá ser convidado para resolver alguma questão durante a aula com o objetivo de validar os exercícios da Lista.

Parte A: Os exercícios desta parte devem ser feitos com base nos exercícios 1, 5, 7, 9, 11, 13, 15 e 18 da Lista 2.

- 1- Apresente o diagrama de classes de projeto refinado com as seguintes notações:
 - a. Dependências estruturais entre as classes de modelo;
 - b. Dependências não estruturais por parâmetro ou variável local entre as classes de controle e modelo;
 - c. Dependências estruturais entre as classes de visão e controle;
 - d. Classes parametrizadas com a estrutura <Set> ou <List> para resolver o lado muitos dos relacionamentos;
 - e. Duas interfaces estabelecendo o devido contrato de comportamento entre as classes consumidoras e fornecedoras.

Parte B: Os exercícios desta parte devem ser feitos com base no exercício 1 desta lista.

- 2- Com base no diagrama de classes de projeto refinado nesta lista, modele o padrão de projeto Composite. Qual o propósito desse padrão no diagrama?
- 3- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Composite.
- 4- Com base no diagrama de classes de projeto refinado nesta lista, modele o padrão de projeto Observer. Qual o propósito desse padrão no diagrama?
- 5- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Observer.
- 6- Com base no diagrama de classes de projeto refinado nesta lista, modele o padrão de projeto Strategy. Qual o propósito desse padrão no diagrama?
- 7- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Strategy.
- 8- Com base no diagrama de classes de projeto refinado nesta lista, modele o padrão de projeto Factory Method. Qual o propósito desse padrão no diagrama?

- 9- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Factory Method
- 10- Com base no diagrama de classes de projeto refinado nesta lista, modele o padrão de projeto Front Controller. Qual o propósito desse padrão no diagrama?
- 11- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Front Controller.
- 12- Com base no diagrama de classes de projeto refinado nesta lista,, modele os pacotes (subsistemas) e faça a alocação das classes em cada pacote. Cada pacote deve mostrar as classes detalhadas com atributos e métodos. Neste exercício, deve constar um pacote de classes de visão, um pacote de classes de controle, no mínimo três pacotes de classes de modelo e um pacote de classes enumeradas.
- 13- Construa o pacote de Persistência e faça a alocação das classes DAO no pacote. Este pacote deve mostrar as classes detalhadas com métodos.
- 14- Apresente a estrutura básica de código para implementar o pacote de Persistência (DAO).
- 15- Após a identificação dos pacotes (subsistemas) e alocação das classes, modele um diagrama de pacotes com os devidos relacionamentos, aplicando o padrão de projeto Façade no pacote de controle e nos pacotes de modelo. Neste diagrama, os pacotes devem mostrar somente os nomes das classes, sem a necessidade de apresentar os detalhes (atributos e/ou métodos) das classes.
- 16- Apresente a estrutura básica de código em JAVA, C# ou C++ para implementar o padrão de projeto Façade.

Parte C: Os exercícios desta parte devem ser feitos com base no exercício 12, 13, 14 e 15 desta lista.

- 17- Faça a alocação dos pacotes (subsistemas) nas camadas de software apresentadas em aula. As camadas devem ser representadas no sentido vertical e com arquitetura aberta.
- 18- Modele um componente para gerenciar o pagamento por meio de cartão, a emissão da nota fiscal eletrônica e a validação do CPF do cliente especial, lembrando que esses componentes são serviços terceirizados e que podem ter sido desenvolvidos numa plataforma diferente.
- 19- A partir da visão dos pacotes (subsistemas) e dos componentes de terceiros, construa o diagrama de componentes. Neste exercício, o pacote de classes enumeradas não precisa ser transformado para um componente e as classes de controle do pacote de controle podem ficar com seu respectivo pacote de classes de modelo, no mesmo componente.
- 20- Com base na alocação dos pacotes (subsistemas) nas camadas de software e no diagrama de componentes, construa o diagrama de implantação distribuindo os componentes em seus respectivos nós. O seu projeto tem quantas camadas? Justifique a tua resposta.

Parte D: Este exercício é de abstração e independente do projeto de sistema da farmácia.

- 21- Abstraia o Mapa Mundi e modele um diagrama de pacotes com os devidos relacionamentos. Somente o nome de cada classe alocada no devido pacote é suficiente para este exercício.