

TRABALHO 2

PROGRAMAÇÃO ORIENTADA A OBJETOS

Requisitos do trabalho.

- Deve ser usada a linguagem Java.
- Devem ser usadas as ferramentas: *NetBeans*, *Javadoc*, *JUnit*, *Cobertura* e *Checkstyle*.
- Não pode ser gerado código automaticamente.
- Os documentos devem ser preenchidos com clareza e cuidados devem ser tomados com a ortografia.
- Devem ser seguidas as convenções de codificação da Sun: <http://java.sun.com/docs/codeconv/>.
- Forneça todos os artefatos produzidos em um CD adequadamente organizado.
- Forneça textos e diagramas no formato PDF.

Questão 1 – 3 pontos

- Crie um diagrama com os módulos nas camadas de apresentação e negócios.
- Descreva cada módulo via um cartão CRC.
- Identifique e documente os contratos entre os módulos.
- Para cada módulo nas camadas de apresentação e negócios, crie um diagrama de classes.
- As classes na camada de apresentação devem ser telas, controladoras de interação, entidades, etc.
- As classes na camada de negócio devem ser controladoras de negócio, comandos, etc.
- As controladoras de negócio criam comandos que serão executados pela camada de persistência.
- A comunicação entre as camadas de negócio e de persistência ocorrerá via o padrão de projeto *Command*.
- Cada comando conterá um método com o código SQL a ser executado para o acesso ao banco de dados.
- Os comandos devem ser organizados em uma hierarquia de classes.
- Os comandos são instanciados pela camada de negócio e passados para a camada de persistência.
- Documente cada classe via um cartão CRC.
- Identifique e documente os contratos entre as classes nos módulos.
- Projete um banco de dados relacional para o armazenamento dos objetos persistentes.
- Documente o banco de dados via um diagrama.

Questão 2 – 3,5 pontos

- Codifique as classes nos módulos da camada de apresentação.
- O modelo de tratamento de eventos deve ser o 1.1.
- O tratamento de eventos deve ser feito através de classes internas (*inner class*).
- Codifique as *interfaces* da camada de negócio.
- Projete e codifique classes *stub* que disponibilizem as *interfaces* da camada de negócio.
- Através das classes *stub* deve ser possível testar a camada de apresentação.
- A *API* pode ser *AWT* ou *Swing*.
- Para cada classe, forneça documentação no formato *Javadoc* e relatório produzido pelo *Checkstyle*.

Questão 3 – 3,5 pontos

- Codifique as classes nos módulos da camada de negócio.
- Codifique as interfaces da camada de persistência.
- Projete e codifique classes *stub* que disponibilizem as *interfaces* da camada de persistência.
- Através das classes *stub* deve ser possível testar a camada de negócio.
- Usando o framework *JUnit*, codifique testes de unidade para os módulos na camada de negócio.
- Os testes de cada unidade devem atingir, no mínimo, a cobertura de 90% das linhas do código sendo testado.
- Forneça descrições de como usar os programas de teste codificados.
- Forneça cópias dos relatórios de cobertura produzidos pela ferramenta *Cobertura*.
- Para as diversas classes, forneça a documentação no formato *Javadoc* e relatório produzido pelo *Checkstyle*.