

# Trabalho Final (2013/02)

## Disciplina de Programação de Software Aplicado

**A) Data de entrega: 02/12/2013**

**Datas de apresentação: 02/12/2013 e 04/12/2013**

### **B) Objetivo:**

O objetivo é consolidar o conhecimento sobre conceitos e construção de sistemas orientados a objetos em arquiteturas multicamadas através da exploração dos tópicos discutidos na disciplina de Programação de Software Aplicado.

### **C) Enunciado do problema:**

Pipocao.com é um gerenciador de coleção de filmes on-line.

O site permite aos seus usuários as seguintes funcionalidades:

- Cadastrar coleções de filmes (tanto em mídias físicas quanto digitais). As operações CRUD completas devem ser disponibilizadas.
- Obter informações cadastrais de filmes a partir do "The Movie Database". Informações sobre o TMDb em:
  - <http://www.themoviedb.org/>
  - <http://www.themoviedb.org/documentation/api>
  - <http://www.themoviedb.org/documentation/api/wrappers-libraries>
- Realizar a crítica de filmes com notas (de 1 a 5) e breves comentários de texto.
- Consultar ranking de críticas dos filmes cadastrados pelos usuários do site.
- Tarefa Bônus: compartilhar no Facebook a crítica de um filme realizada pelo usuário.

### **D) Requisitos:**

Os seguintes itens são obrigatórios na implementação do sistema:

- Arquitetura multicamada (pelo menos 3) explorando os padrões vistos em aula;
- Interface gráfica de usuário via web;
- Persistência em banco de dados relacional;
- Utilização de pelo menos um framework;
- Disponibilização de pelo menos uma das funcionalidades via web services.

Os seguintes requisitos adicionais serão avaliados na execução do trabalho:

- Qualidade da independência entre camadas;
- Correta implementação do conceito de transação nas operações de acesso ao banco de dados;
- Correta implementação do tratamento de exceções;
- Correta implementação do tratamento da entrada de dados com validação das informações.

### **E) Desenvolvimento, apresentação e avaliação do trabalho:**

- O trabalho pode ser realizado individualmente ou em grupos de, no máximo, 4 alunos.
- O desenvolvimento do trabalho pode ser realizado em qualquer ambiente de programação e linguagem orientada a objetos.
- Os trabalhos serão apresentados no laboratório. Durante a apresentação, TODOS os alunos devem estar presentes e aptos a responder as perguntas. Respostas insatisfatórias por um aluno ou a sua ausência acarretarão descontos na nota final. A apresentação do trabalho é de inteira responsabilidade dos alunos (configuração da máquina, do ambiente de software, banco de dados, etc.) e o código-fonte

utilizado deverá ser o mesmo entregue ao professor. É tarefa do grupo garantir que o sistema esteja apto a ser executado no dia da apresentação.

- Sistemas que não consigam ser executados ou apresentados no dia da apresentação receberão nota zero.
- Mensagens de erro apresentadas durante a execução do programa, mesmo que a aplicação não pare de executar, serão consideradas como erros de execução.
- Em caso de erro de sintaxe (compilação), o peso final do trabalho será valorado em zero.
- Em caso de erro de semântica (conteúdo), o peso final do trabalho sofrerá uma redução.
- Os trabalhos serão avaliados de acordo com critérios a serem estabelecidos pelo professor da disciplina, considerando o que é pedido no enunciado e o que foi realizado com sucesso pelo sistema. Também serão avaliadas a modelagem do sistema (correta criação das classes necessárias, com seus atributos e métodos, encapsulamento, e correto estabelecimento de relações entre as classes) e sua implementação de acordo com os conceitos de orientação a objetos e arquitetura multicamada.
- ***Trabalhos copiados resultarão em nota zero para todos os alunos envolvidos.***

#### F) Entrega do trabalho:

- Todos os arquivos necessários a execução do sistema, bem como os arquivos-fonte e os arquivos de documentação, deverão ser empacotados em um único arquivo (.zip) e submetidos através do sistema Moodle até a data de entrega.
- Devem fazer parte da documentação (documento texto):
  - Diagrama de classe do sistema.
  - Diagrama relacional da base de dados.
  - Diagrama de pacotes (mostrando a arquitetura multicamada do sistema).
  - Os diagramas devem estar disponíveis em imagens com resolução suficiente e de fácil visualização. Não serão aceitos diagramas que estejam em formato original da ferramenta de desenho (como Visio, Astah, e outros).
- Não serão aceitos trabalhos enviados por correio eletrônico.
- Não serão aceitos trabalhos enviados fora do prazo estabelecido.