



Sistema de Software de Museu 2.0

Trabalho 2 – Programação Orientada a Objetos

Prof. André Marcelo Panhan

Conteúdo

1.	Introdução.....	3
2.	Arquitetura	3
2.1	Visão (Apresentação)	4
2.2	Controle	4
2.3	Modelo.....	4
3.	Domínio do Sistema	4
3.1	Requisitos Funcionais.....	5
3.2	Especificação Textual.....	6
3.3	Banco de Dados	7
3.4	Tratamento de Exceção	7
3.5	Padrões de Projetos.....	7
3.6	Documentação	7
4.	Cronograma.....	8
4.1	Datas	8
5.	Conclusão	8

1. Introdução

Este documento apresenta o projeto do “**Sistema de Software de Museu 2.0**”, um sistema desenvolvido como estudo de caso na atividade da disciplina de Programação Orientada a Objetos (POO) no IFSP – Bragança Paulista. O sistema deverá ser desenvolvido em *JAVA* utilizando o *framework Java Foundation Classes* (JFC).

Este documento é dividido em quatro seções primárias. A primeira seção apresenta o contexto deste do projeto para a disciplina POO. A segunda seção apresenta as características da arquitetura do sistema, padrões de projetos e as tecnologias que serão empregadas no desenvolvimento. Já a terceira seção apresenta os Requisitos Funcionais, Regras de Negócios, Banco de Dados, Tratamento de Exceções e a Documentação que deverão ser utilizados. Por fim, a seção quatro apresenta um cronograma para desenvolvimento do projeto.

2. Arquitetura

O “**Sistema de Software de Museu 2.0**” deverá ser desenvolvido em uma arquitetura em três camadas, as camadas básicas serão Visão, Controle e Modelo.

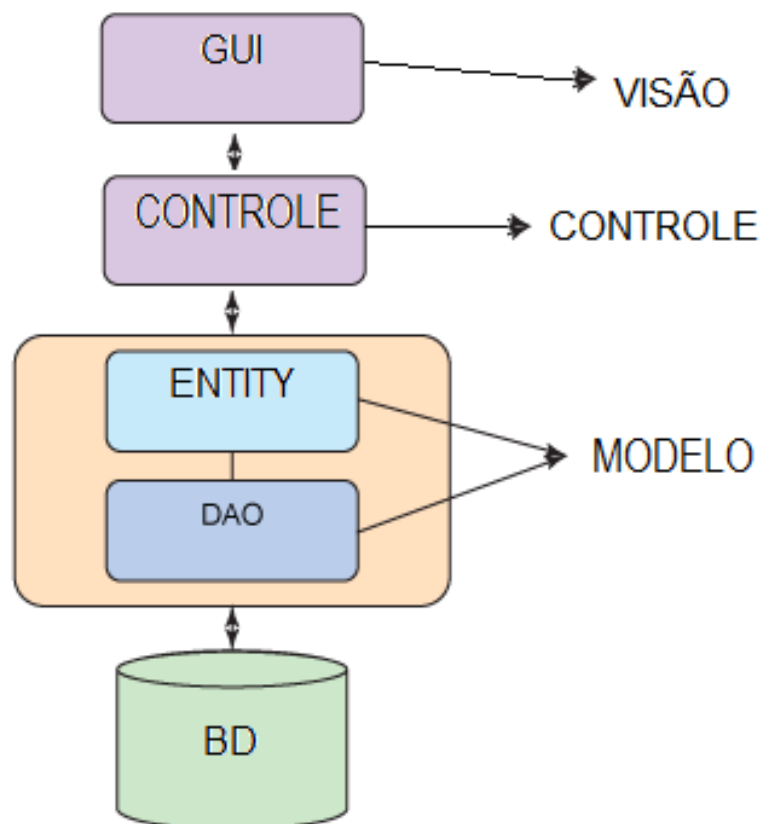


Figura 1 - Arquitetura do Sistema

2.1 Visão (Apresentação)

A **camada de visão** (*view*) deverá ser desenvolvida utilizando o *framework* Java Foundation Classes (JFC). O Java Foundation Classes (JFC) é um framework gráfico para construção de aplicações Java desktop. O JFC consiste principalmente nas bibliotecas Abstract Window Toolkit (AWT) e Java Swing API.

A biblioteca AWT é um conjunto básico de classes e interfaces que definem os componentes de uma janela desktop. A biblioteca AWT é a base para Java Swing API.

A Java Swing API é um conjunto sofisticado de classes e interfaces que definem os componentes visuais e serviços necessários para construir uma interface gráfica de usuário.

2.2 Controle

A **camada de controle** (*controller*) é responsável por enviar comandos para o modelo para atualizar o seu estado. O controlador também pode enviar comandos para a visão associada para alterar a apresentação da visão do modelo.

2.3 Modelo

A **camada de modelo** (*model*) armazena dados e notifica seus controladores associados quando há uma mudança em seu estado. Estas notificações permitem que as visões produzam saídas atualizadas e que os controladores alterem o conjunto de comandos disponíveis.

A **camada de modelo** deverá ser desenvolvida utilizando o padrão de projetos Data Access Object (DAO). O padrão DAO permite criar as classes de dados independentemente da fonte de dados ser um BD relacional, um arquivo texto, um arquivo XML, etc. Para isso, ele encapsula os mecanismos de acesso a dados e cria uma interface de cliente genérica para fazer o acesso aos dados permitindo que os mecanismos de acesso a dados sejam alterados independentemente do código que utiliza os dados.

3. Domínio do Sistema

O “**Sistema de Software de Museu 2.0**” deverá ser desenvolvido em linguagem de programação *JAVA*, utilizando o paradigma orientado a objetos, realizando a gestão das visitas, acervo, exposição, restauração, eventos e venda de souvenir. O sistema deve ser capaz de controlar um Museu, permitindo o controle do fluxo de visitantes; mantendo um cadastro unificado de acervo físico e virtual; controlando as exposições, temporárias ou permanentes; gerenciando o processo de restauração de obras; mantendo informações sobre os eventos a serem realizados; gerenciando a loja de souvenirs. Esta

seção descreve detalhadamente os requisitos funcionais necessários para desenvolvimento do sistema.

3.1 Requisitos Funcionais

O “**Sistema de Software de Museu 2.0**” deverá possuir um número mínimo de requisitos funcionais.

Desta forma, o grupo de trabalho deverá utilizar os conceitos de Orientação a Objetos para desenvolver este sistema, respeitando a arquitetura proposta na seção 2.

Abaixo, estão relacionados os Requisitos Funcionais que deve estar presente no sistema:

- RF01: Manter controle de fluxo de visitantes do museu através de um cartão magnético fornecido para o visitante na entrada.
- RF02: Manter o cadastro dos dados básicos de cada visitante para envio de convite de exposição ou evento.
- RF03: Manter o cadastro unificado de acervo físico (obras em geral) e virtual (Documentos Históricos).
- RF04: Manter o cadastro de exposições, temporárias ou permanentes do museu, permitindo listar as obras exibidas em cada exposição, além de mostrar a salas reservadas e o máximo de visitantes.
- RF05: Manter o processo de restauração de obras, desde a abertura da solicitação de serviço, duração e andamento até a conclusão da restauração da obra.
- RF06: Manter o cadastro de evento a ser realizado, assim como quem será o responsável e o número de visitantes.
- RF07: Manter o controle de estoque da Loja de Souvenirs, desde a aquisição de novos produtos, vendas até o fornecimento de relatórios de vendas para o gerente deste setor.

Para desenvolvimento dos Requisitos Funcionais o grupo de trabalho deverá atender as seguintes Regras de Negócios:

- RN01: Todos os visitantes deverão ser cadastrados.
- RN02: Não poderão ser incluídas em exposições temporárias ou permanentes obras que estejam no processo de restauração no mesmo período.
- RN03: Não poderão ser incluídas em exposições temporárias ou permanentes salas que já estejam reservadas no mesmo período.
- RN04: As obras físicas ou virtuais deverão formar o cadastro único do acervo do Museu.

3.2 Especificação Textual

Gestão de Visita (CSU1)

Descrição: Gerencia o fluxo de visitantes do museu por meio de um cartão magnético fornecido para o visitante na entrada, sendo possível armazenar os dados básicos de cada visitante para envio de convite de exposição ou evento.

Gestão de Acervo (CSU2)

Descrição: Gerencia as obras presentes no museu por meio de um cadastro unificado de acervo físico (obras em geral) e virtual (Documentos Históricos), onde poderão ser realizadas consultas das obras.

Gestão de Exposição (CSU3)

Descrição: Gerencia as exposições, temporárias ou permanentes do museu, permitindo listar as obras exibidas em cada exposição, além de mostrar a salas reservadas e o máximo de visitantes.

Gestão de Restauração (CSU4)

Descrição: Gerencia o processo de restauração de obras, desde a abertura da solicitação de serviço, duração e andamento até a conclusão da restauração da obra.

Gestão de Evento (CSU5)

Descrição: Gerencia as informações do evento a ser realizado, assim como quem será o responsável e o número de visitantes para o evento em questão.

Gestão de Venda de Souvenir (CSU6)

Descrição: Gerencia a loja de souvenirs, desde o seu estoque até a aquisição de novos produtos para a loja. Fornece relatórios de vendas para o gerente deste setor.

Para um melhor entendimento do domínio da aplicação, o estudante deve realizar visitas virtuais nos diversos museus nos endereços:

http://www.saopaulo.sp.gov.br/conhecasp/cultura_museus

<http://www.louvre.fr/en>

3.3 Banco de Dados

O grupo de trabalho deverá desenvolver/atualizar o Modelo Entidade Relacionamento (MER) do “**Sistema de Software de Museu 2.0**”, baseado na estrutura e relações das classes de negócios que serviram de modelo para objetos do sistema.

O Banco de Dados desenvolvido no SGBD *MySQL*, deverá refletir os conceitos de orientação a objetos.¹

3.4 Tratamento de Exceção

A linguagem Java tem uma abordagem própria para lidar com erros de execução. Na abordagem Java a classe **Throwable** modela todos os tipos de *erros de execução*. Há duas subclasses de *Throwable*: *Error* e *Exception*.

O grupo de trabalho deverá utilizar como controle de exceção as subclasses de *Exception* que definem erros para os quais as aplicações normalmente têm condições de definir um tratamento e exibir mensagens de erro com *JOptionPane*.

3.5 Padrões de Projetos

Um padrão de projeto é uma solução consolidada para um problema recorrente no desenvolvimento e manutenção de software orientado a objetos. Os padrões de projetos são denominados padrões “*Gang of Four*” (GoF). Eles são classificados em três categorias: padrões de criação, estruturais e comportamentais.

O grupo de trabalho deverá escolher dois padrões de projetos de criação, estruturais ou comportamentais e utilizá-los no desenvolvimento do **Sistema de Software de Museu 2.0**².

3.6 Documentação

O **Javadoc** é um gerador de documentação criado pela Sun Microsystems para documentar a API dos programas em Java, a partir do código-fonte. O resultado é expresso em HTML. É constituído, basicamente, por algumas marcações simples inseridas nos comentários do programa.

O grupo de trabalho deverá elaborar a documentação do **Sistema de Software de Museu 2.0** utilizando o **Javadoc**. Para a elaboração da documentação do sistema o grupo de trabalho deve observar que toda documentação criada deve sofrer uma

¹ De todos os conceitos de orientação a objetos, um dos mais complexos de se modelar é o de Herança.

² Os Padrões de Projeto: *Factory Method*, *MVC* e *DAO* já compõem os requisitos do Sistema de Software de Museu 2.0 e não poderão ser escolhidos pelos Grupos de Trabalho.

atualização periódica, de acordo com as mudanças ocorridas, para que a mesma possa refletir a realidade dos fatos.

4. Cronograma

Esta última seção descritiva detalha as etapas do desenvolvimento do “**Sistema de Software de Museu 2.0**”.

4.1 Datas

Apresentação do **Sistema de Software de Museu 2.0**: 21/11/2017

5. Conclusão

O projeto apresentado cobre as etapas de desenvolvimento do “**Sistema de Software de Museu 2.0**”, um sistema desenvolvido com o *framework* Java Foundation Classes (JFC). O desenvolvimento dele coincidirá com o andamento da disciplina Programação Orientada a Objetos no IFSP, servindo como exemplo de implementação.