



Java Enterprise Edition - JEE

Análise e Projeto Orientado a Objetos com MVC



Esp. Márcio Palheta

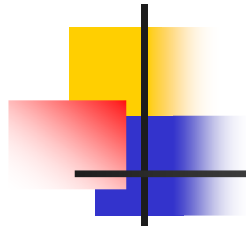
gtalk: marcio.palheta@gmail.com

marcio.palheta@fucapi.br



Ementa:

- Revisão de JEE com MVC;
- Camada de controle com Struts;
- Persistência com Hibernate;
- Arquitetura de software com:
 - Model: JSP, Servlets e Filtros;
 - Controller: Struts;
 - Model: Dao e Hibernate;
- Novos padrões de projeto:
 - Facade, Singleton, ServiceLocator



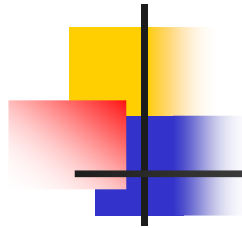
Agenda

- Revisão MVC
- Estudo de caso
- Levantamento de requisitos
- Análise de Sistema
- Projeto de Sistema
- Implementação



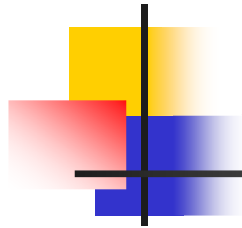
Estudo de caso

- Empresa do ramo de calçados
- Problemas de comunicação;
- Dificuldade para localizar o ramal de um determinado funcionário;
- Sobrecarga de atividades da telefonista;
- Como resolver?



Levantamento de requisitos

- Identificação de usuários envolvidos
- Entrevistas
- Definição da visão inicial
- Identificação de requisitos funcionais
- Escopo negativo
- Diagrama de contexto



Análise de sistemas

- Analisar casos de uso
 - Encontrar classes iniciais e distribuir comportamentos dos casos de uso entre elas;
 - Para cada classe, descrever as responsabilidades, atributos e associações;
 - Classes de fronteira, controle e entidade

Diagrama de casos de uso

- O administrador do sistema deve cadastrar todos os funcionários da empresa;

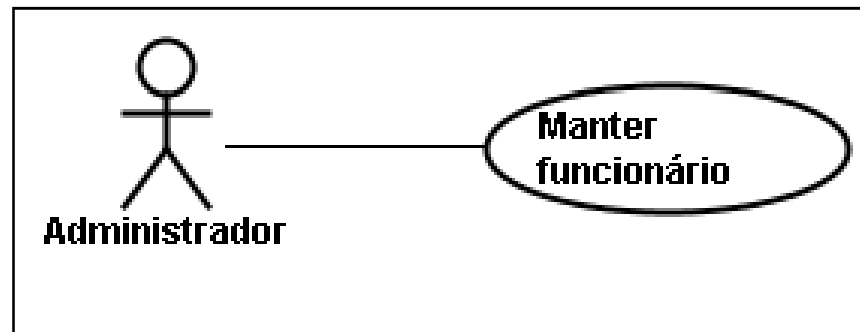
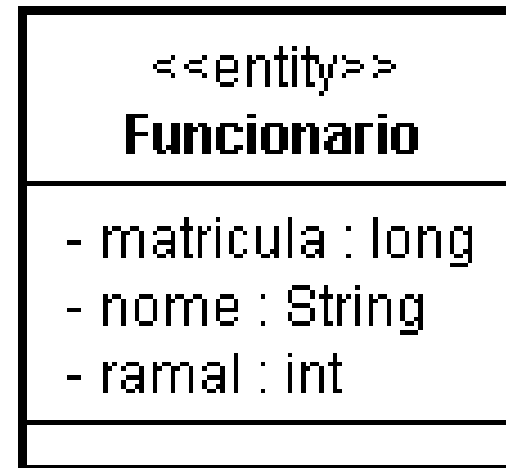


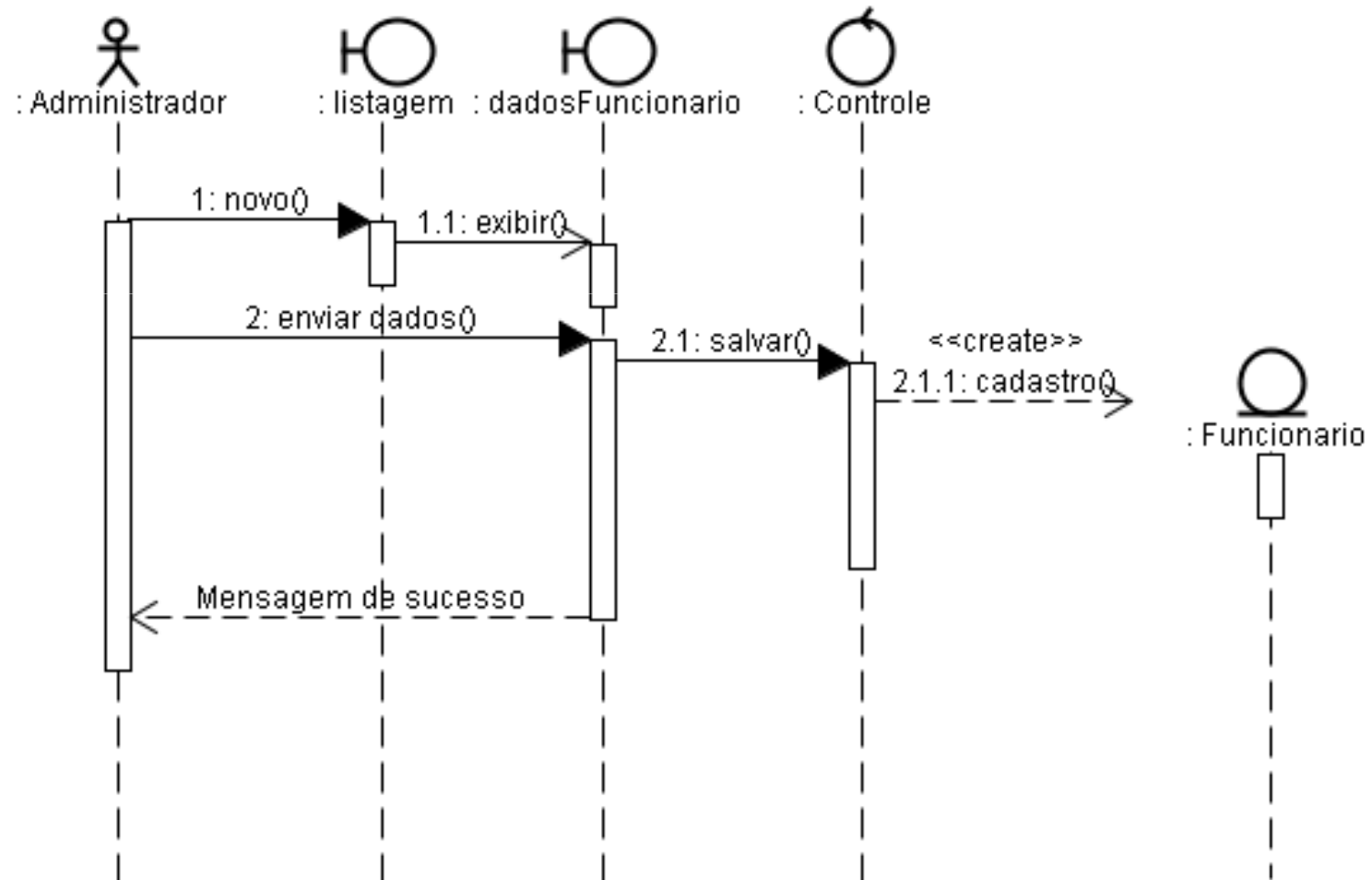


Diagrama de classes

- Diagrama de estrutura
- Descrever as classes entidade que existem no sistema, contendo:
 - Atributos
 - Métodos
 - relacionamentos



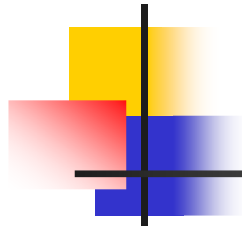
Análise: Cadastrar Funcionário





Projeto de sistemas

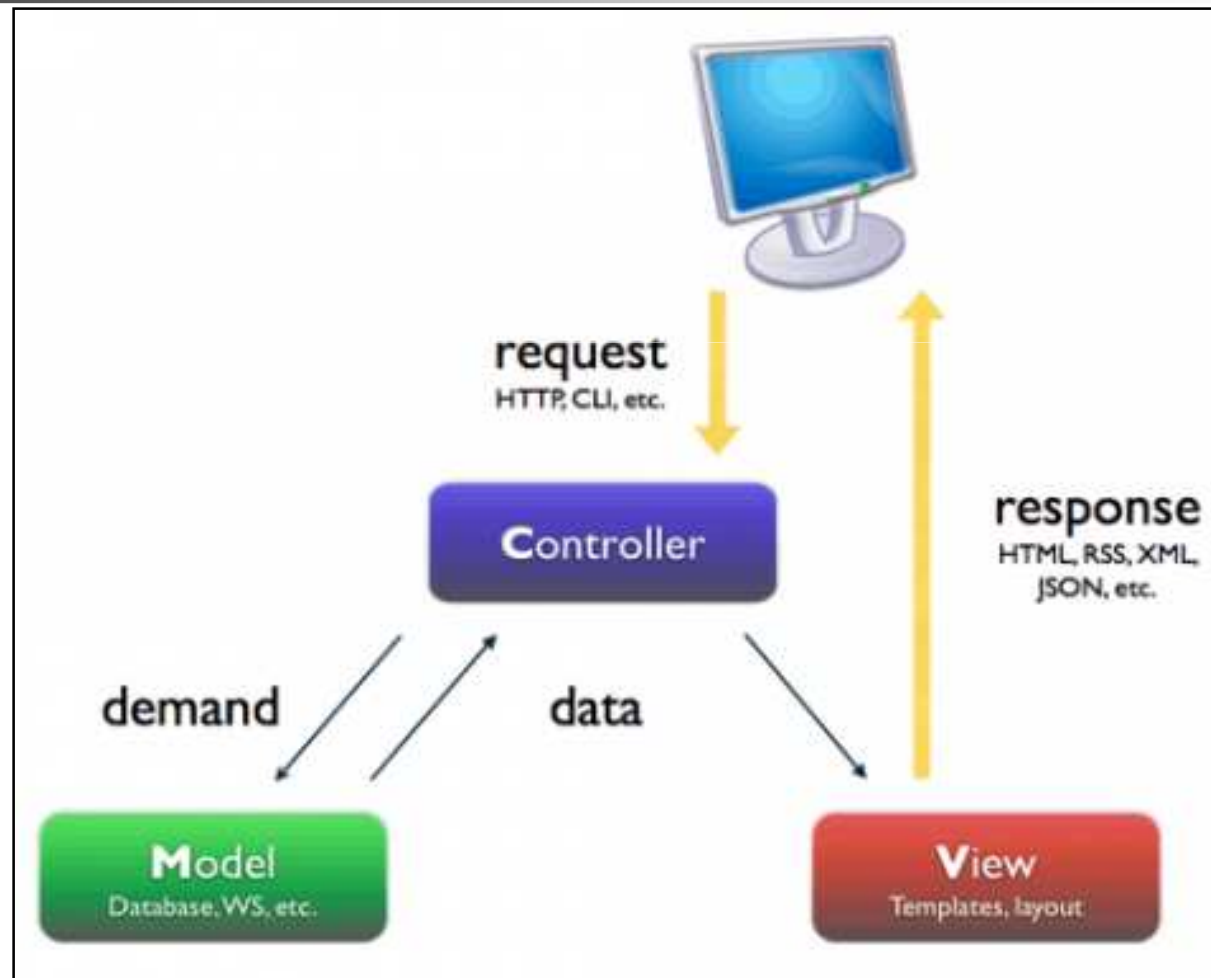
- Projetar arquitetura
 - Avaliar elementos da fase de análise;
 - Definir elementos da fase de projeto;
 - Organizar os novos elementos em pacotes;
 - Definir estrutura da aplicação



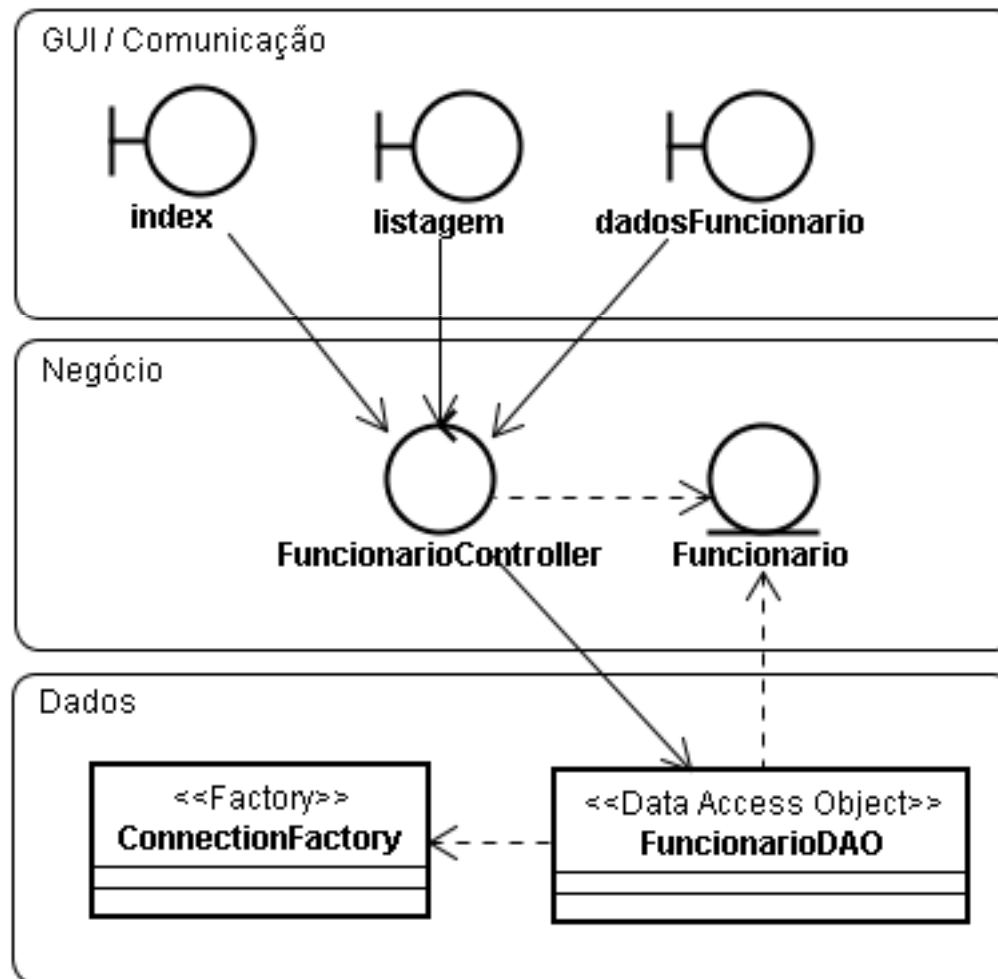
Distribuição em pacotes

- br.fucapi.revisao.model.bean
 - Funcionario.java
- br.fucapi.revisao.model.connection
 - ConnectionFactory.java
- br.fucapi.revisao.model.dao
 - FuncionarioDAO.java
- br.fucapi.revisao.controller
 - FuncionarioController

Padrão de projeto - MVC



Arquitetura simplificada





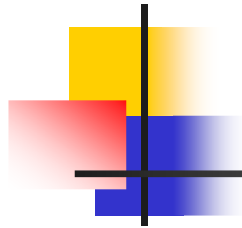
Modelo de dados

```
Create database cadastrofuncionario;  
use cadastrofuncionario;  
Create Table funcionario(  
    matricula int not null auto_increment,  
    nome varchar(50),  
    ramal int,  
    primary key (matricula)  
);
```



Lista de atividades

- Criação da base de dados;
- Criação do projeto `sistemafuncionario`;
- Criação da camada MODEL;
 - `Funcionario`, `ConnectionFactory` e `FuncionarioDAO`;
- Criação da camada CONTROLLER;
 - `FuncionarioController`
- Criação da camada VIEW;
 - Páginas JSP



O que vem a seguir?

- Filter API;
- Camada de controle com struts;
- Persistência com Hibernate;
- Arquitetura de software com:
 - View: JSP, Servlets e Filtros;
 - Controller: Struts;
 - Model: Dao e Hibernate;
- Novos padrões de projeto:



Referências

- Hall, Marty, "Core Servlets and Java Server Pages", Janeiro 2002, Sun Microsystems Press;
- <http://java.sun.com/>
- <http://java.sun.com/j2ee/1.6/docs/tutorial/doc/index.html>
- <http://java.sun.com/products/jndi/docs.html>
- <http://java.sun.com/blueprints/corej2eepatterns/Patterns/index.html>



Java Enterprise Edition - JEE

Análise e Projeto Orientado a Objetos com MVC



Esp. Márcio Palheta

gtalk: marcio.palheta@gmail.com

marcio.palheta@fucapi.br