

Java Server Faces

Modelo MVC e Estrutura das Aplicações


www.softblue.com.br

Todos os direitos de cópia reservados. Não é permitida a distribuição física ou eletrônica deste material sem a permissão expressa e por escrito do autor.

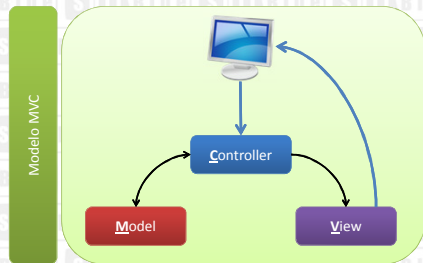
Tópicos Abordados

- O modelo MVC
 - Controller
 - View
 - Model
- O MVC e as aplicações JSF
- Estruturando a camada model
- Entidades
- Acesso a dados
- Definição dos objetos de negócio

O Modelo MVC

- **Model – View – Controller**
- Permite a divisão da aplicação em camadas lógicas
- Esta divisão facilita a manutenção e extensão futura do código

Esquema do MVC



Controller

- É o componente chamado pelo cliente
- O controller executa os seguintes passos:
 1. Prepara os dados que serão usados pelo **model**
 2. Chama o **model**
 3. Prepara os dados que serão usados pela **view**
 4. Direciona para a **view**

View

- É o resultado visual observado pelo cliente
- O direcionamento para uma determinada **view** é responsabilidade do **controller**
- Os dados presentes na **view** normalmente são renderizados com base em dados obtidos pelo **model**

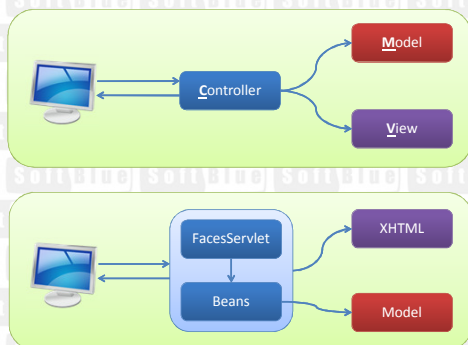
Model

- É onde está toda a lógica de negócio implementada pela aplicação
- O model compreende implementação da lógica, acesso a bancos de dados, acesso a sistemas externos, etc.
- O model não deve ter qualquer dependência com a tecnologia usada pelos controllers ou views
 - Esta independência garante que o mesmo model possa ser usado em qualquer tipo de aplicação (web, desktop, dispositivos móveis, etc.)

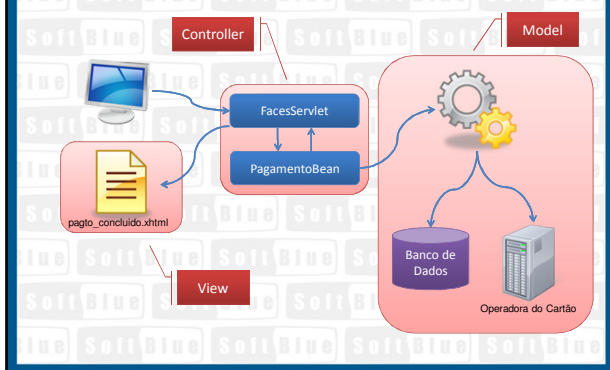
O MVC e as Aplicações JSF

- O MVC pode ser aplicado para desenvolver qualquer tipo de aplicação
- Quando a aplicação é web, o MVC se encaixa perfeitamente
 - O modo de funcionamento proposto pelo MVC e das aplicações web é muito similar
- No caso do JSF, o próprio framework implementa boa parte da camada controller

O MVC e as Aplicações JSF



Exemplo de Aplicação do MVC



Estruturando a Camada Model

- O model é, sem dúvida, a camada mais importante da sua aplicação
- Nela está localizada a lógica de todo o funcionamento da aplicação
- É preciso que o model seja bem organizado para que o código seja de fácil manutenção e de fácil extensão
- Relembrando: a camada model deve ser independente da tecnologia utilizada

Definição das Entidades

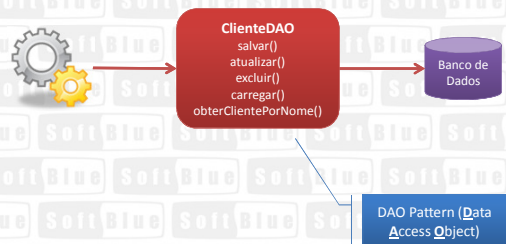
- As entidades (**entities**) são classes que representam os conceitos presentes na aplicação
- Na prática elas normalmente são mapeadas para tabelas do banco de dados



Acesso a Dados

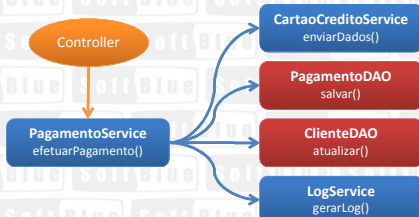
- Praticamente todas as aplicações comerciais possuem um meio de acessar e armazenar dados
 - Banco de dados
 - Sistemas externos
 - Arquivos
- É importante criar um conjunto de classes que encapsulem o acesso aos dados
 - Centraliza o acesso aos dados
 - Permite mudar o mecanismo de persistência sem afetar outras partes do código

Acesso a Dados

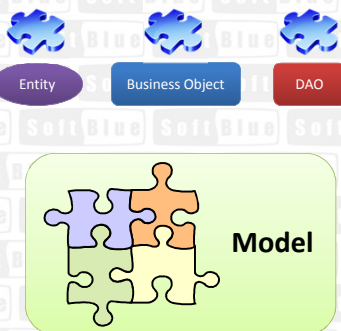


Definição dos Objetos de Negócio

- Os objetos de negócio (**business objects** ou **services**) realizam tarefas específicas de negócio



O Model é uma Composição



Colocando em Prática...



Agora que você já aprendeu a teoria, acesse as vídeo-aulas práticas e pratique os assuntos abordados neste módulo!

[Clique aqui para acessar as vídeo-aulas práticas](#)
