

PROGRAMAÇÃO PARA WEB I

AULA 14

Profa. Silvia Bertagnolli

INTEGRANDO SERVLETS E JSP

PASSOS PARA INTEGRAR SERVLETS E JSP

Passo 1 – Mapear as transições entre as páginas

Passo 2 – Criar os pacotes model, controller e pojo

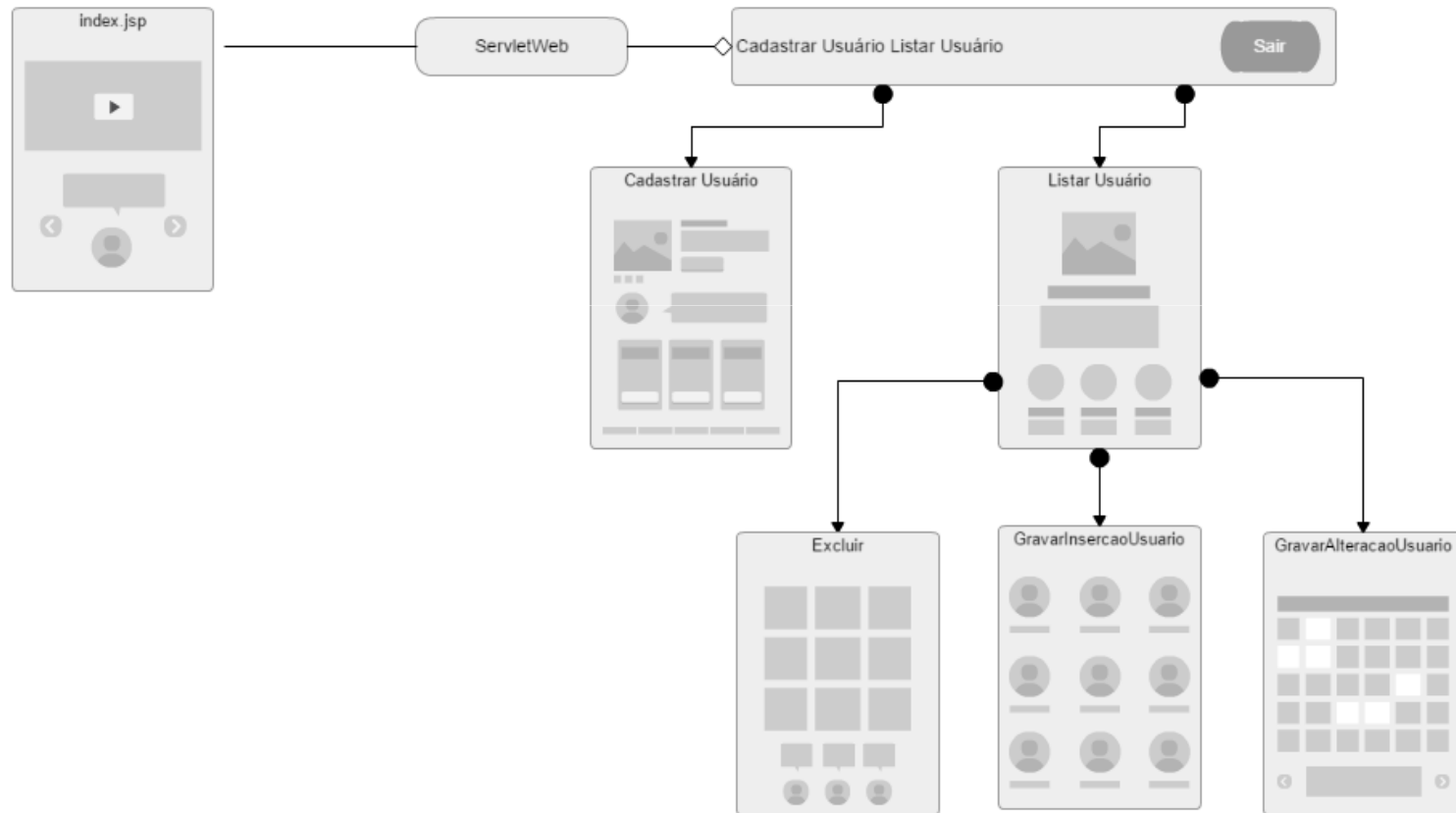
Passo 3 – Criar o servlet no pacote controller e mapear as ações

Passo 4 – Criar uma classe Controller para o POJO

Passo 5 – Criar as páginas JSP

PASSO 1

PASSO 1 — MAPEAR AS TRANSIÇÕES ENTRE AS PÁGINAS



PASSO 2

CRIAR OS PACOTES MODEL, CONTROLLER E POJO

Criar a classe Usuário no pacote POJO

Criar as classes e interface do pacote model:

- Classe ConnectionFactory, que estabelece a conexão com o BD
- Interface GenericDAO

```
public interface GenericDAO <T> {  
    public int insert(T obj);  
    public int update(T obj);  
    public int delete(T obj);  
    public java.util.List<T> getAll();  
    public T findById(int id);  
}
```

CRIAR OS PACOTES MODEL, CONTROLLER E POJO

Criar a enumeração de SQLs:

```
public enum SQLs {  
    //USUÁRIO  
    USR_VALIDA("SELECT * FROM Usuario WHERE nome=? AND senha=?"),  
    USR_INSERT("INSERT INTO Usuario(senha, nome, nomeCompleto, email)  
                VALUES(?,?,?,?)"),  
    USR_UPDATE("UPDATE Usuario SET senha=?, nome=?, nomeCompleto=?,  
                email=? WHERE idUsuario = ?"),  
    USR_DELETE("DELETE FROM Usuario WHERE idUsuario = ?"),  
    USR_LISTALL("SELECT * FROM Usuario ORDER BY nome"),  
    USR_FIND_BY_ID("SELECT * FROM Usuario WHERE idUsuario = ?");  
  
    //....  
}
```


CRIAR OS PACOTES MODEL, CONTROLLER E POJO

Criar a classe DAO:

```
public class UsuarioDAO implements GenericDAO<Usuario> {  
  
    ...  
  
}
```

Vamos analisar a classe UsuarioDAO?

PASSO 3

PASSO 3 — CRIAR O SERVLET NO PACOTE CONTROLLER E MAPEAR AS AÇÕES

```
public class ServletWeb extends HttpServlet {  
  
    ...  
  
}
```

Vamos analisar a classe ServletWeb?

PASSO 3 — CRIAR O SERVLET NO PACOTE CONTROLLER E MAPEAR AS AÇÕES

- O que faz o método `init()`?
- Quais outros recursos esse método utiliza?
- Qual a origem dos dados “usuario” e “senha” usados no método `init()`?
- O que fazem os métodos `doPost()`, `doGet()` e `processRequest()`?

PASSO 3 — CRIAR O SERVLET NO PACOTE CONTROLLER E MAPEAR AS AÇÕES

```
public class ServletWeb extends HttpServlet {  
  
    ...  
  
}
```

Vamos analisar a classe ServletWeb?

MÉTODO PROCESSREQUEST()

```
String acao = request.getParameter("acao");  
switch (acao) {  
    case "Logar": jsp = ControllerUsuario.login(request);  
                  break;  
    case "ListarUsuario": jsp = ControllerUsuario.listar(request);  
                          break;  
    ...
```

PASSO 4

PASSO 4 - CRIAR UMA CLASSE CONTROLLER PARA O POJO

```
public class ControllerUsuario {  
    public static String login(HttpServletRequest request) {  
        String erro="";    String jsp="";  
        Usuario usr = new Usuario();  
        usr.setNome(request.getParameter("nome"));  
        usr.setSenha(request.getParameter("senha"));  
        try {  
            usr = new UsuarioDAO().validate(usr);  
            if(usr == null){  
                erro="Usuário não encontrado!";  
                request.setAttribute("erro", erro);  
                jsp = "/erro.jsp";  
            }  
            else{ ... }  
        }  
    }  
}
```


PASSO 4 - CRIAR UMA CLASSE CONTROLLER PARA O POJO

```
public class ControllerUsuario {  
    public static String login(HttpServletRequest request) {  
        ...  
  
        else{  
            request.getSession().setAttribute("idusuario",  
                                                usr.getIdUsuario());  
            jsp = "/menu.jsp";  
        }  
    } catch (Exception e) {  
        e.printStackTrace(); jsp = "";  
    }  
    return jsp;  
}
```

PASSO 4 - CRIAR UMA CLASSE CONTROLLER PARA O POJO

Vamos analisar a classe ControllerUsuario?

O que faz o método validarSessao()?

Qual a função dos métodos listar(), alterar(), excluir(), inserir() e sair()?

Quando são usados os métodos gravarAlteracao() e gravarInsercao()?

PASSO 5

PASSO 5 — CRIAR AS PÁGINAS JSP

Vamos analisar a página `index.jsp`. Observe que a execução do sistema começa por essa página!

Vamos analisar a página `erro.jsp`. Para que ela é usada?

Vamos analisar a página `menu.jsp`.

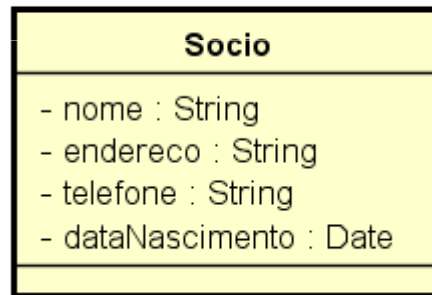
Vamos analisar a página `inserirusuario.jsp`. Como funciona o botão Gravar dessa página?

Vamos analisar a página `listarusuario.jsp`. Como funciona o botão Excluir dessa página? Como funcionam os links exibidos na tabela?

EXERCÍCIOS

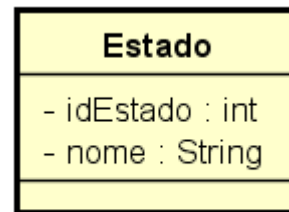
EXERCÍCIOS

Crie um sistema que permite cadastrar, listar e excluir sócios de um clube. Cada sócio é descrito pela classe abaixo:



EXERCÍCIOS

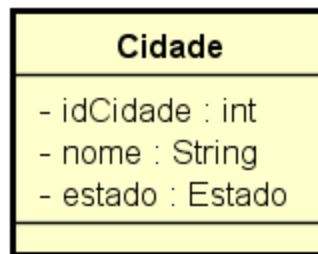
Vamos adaptar o sistema que você começou a desenvolver para cadastrar, que tem como classe básica a classe Estado com os atributos abaixo:



powered by Astah

EXERCÍCIOS

Vamos adaptar o sistema que você começou a desenvolver para cadastrar e listar as cidades, que tem como classe básica a classe Cidade com os atributos abaixo:



EXERCÍCIOS

Vamos criar adaptar o sistema que já existe para possibilitar cadastrar, listar e excluir Sócios, que tem como classe básica a classe Sócio com os atributos abaixo:

