

Linguagem Técnica de Programação Web

Aula 5 – Desenvolvendo uma aplicação JSP com JDBC – Parte 1 (insert)

Prof. João Paulo Pimentel
joao.pimentel@projecao.br

Model-View-Controller (MVC)

Objetivo:

- Separar dados ou lógica de negócios (Model) da interface do usuário (View) e do fluxo da aplicação (Control)
 - A ideia é permitir que uma mesma lógica de negócios possa ser acessada e visualizada através de várias interfaces.
 - Na arquitetura MVC, a lógica de negócios (chamaremos de Modelo) não sabe de quantas e nem quais interfaces com o usuário estão exibindo seu estado.
 - Com as diversas possibilidades de interfaces que conhecemos hoje, a MVC é uma ferramenta indispensável para desenvolvermos sistemas
-

Criando o Banco no MySQL

No MySQL Workbench criar o DATABASE **Login** e a tabela **tb_login**, conforme abaixo:

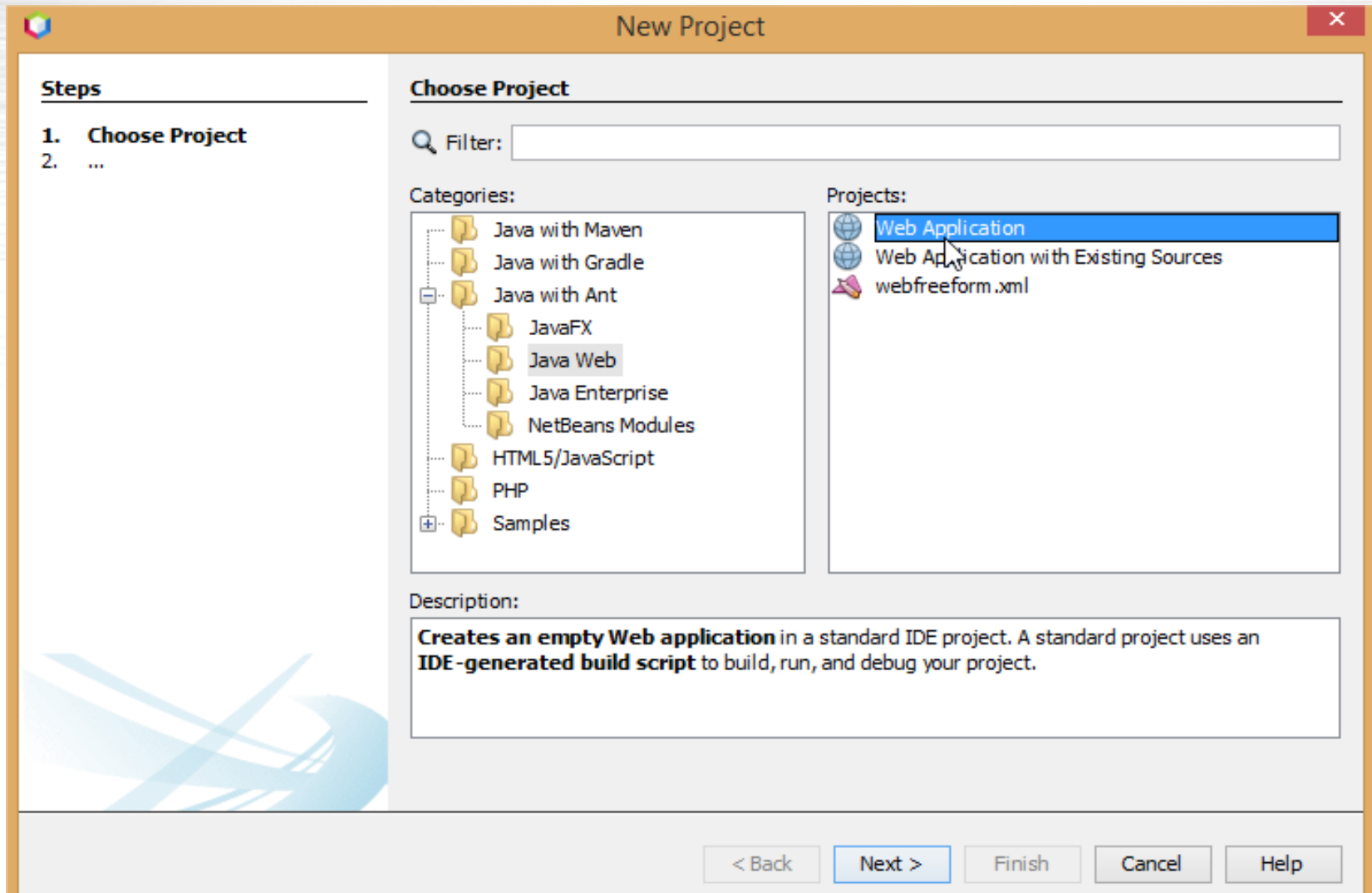
```
CREATE SCHEMA IF NOT EXISTS `Login` DEFAULT CHARACTER SET utf8;  
Use `Login`;
```

```
CREATE TABLE IF NOT EXISTS `Login`.`tb_login` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `email` VARCHAR(50) NOT NULL,  
  `senha` VARCHAR(8) NOT NULL,  
  PRIMARY KEY (`id`))  
ENGINE = InnoDB;
```

```
select * from tb_login;
```

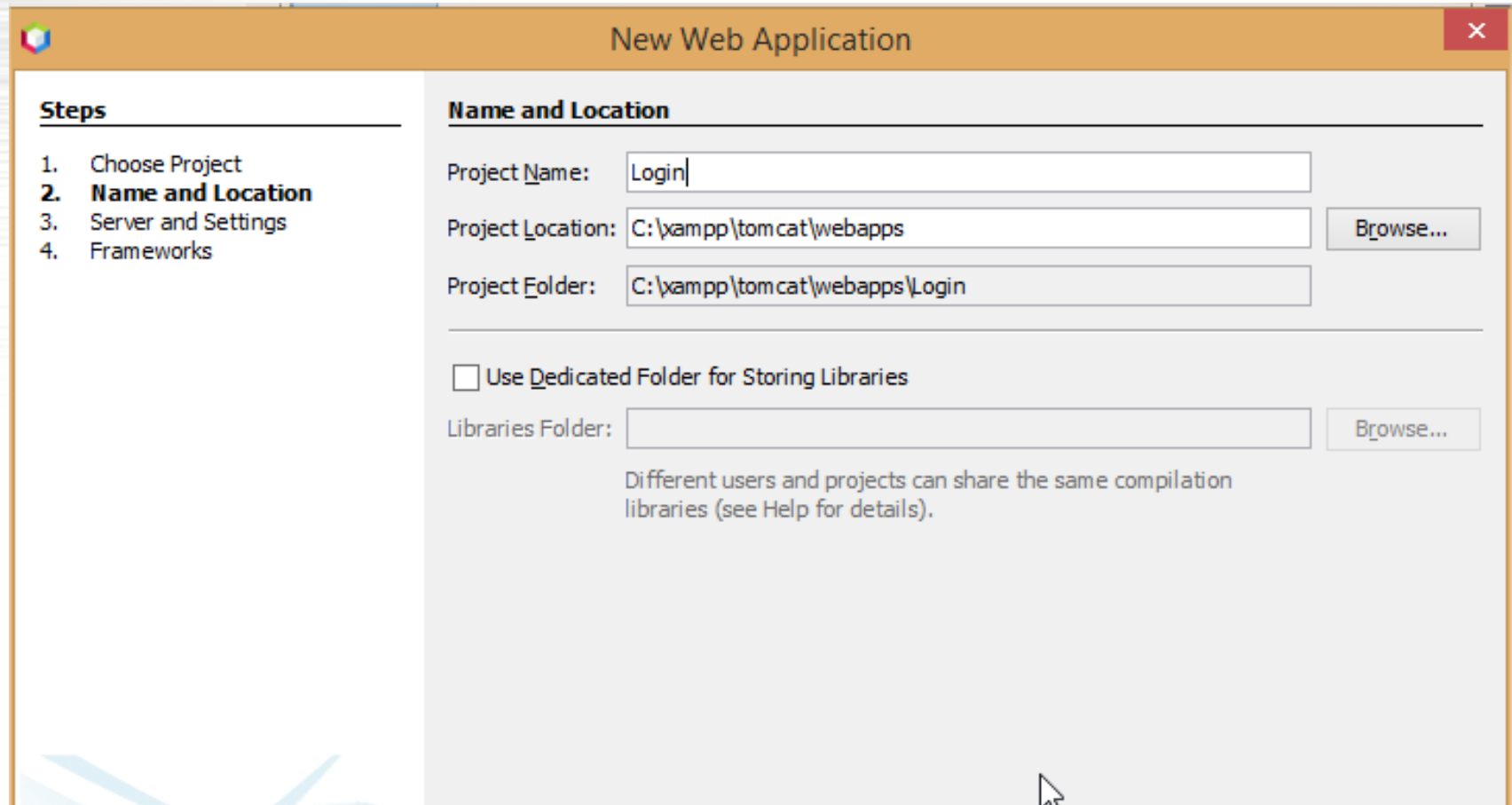
Criando o Projeto no NETBEANS

Criar um novo Projeto Web



Criando o Projeto no NETBEANS

Project Name: Login



The screenshot shows the 'New Web Application' dialog box in NetBeans. The window has an orange title bar with the NetBeans logo on the left and a close button on the right. The main area is divided into two panes. The left pane, titled 'Steps', contains a list of four steps: 1. Choose Project, 2. Name and Location (which is highlighted in bold), 3. Server and Settings, and 4. Frameworks. The right pane, titled 'Name and Location', contains three text input fields: 'Project Name' with the value 'Login', 'Project Location' with the value 'C:\xampp\tomcat\webapps', and 'Project Folder' with the value 'C:\xampp\tomcat\webapps\Login'. To the right of the 'Project Location' and 'Project Folder' fields are 'Browse...' buttons. Below these fields is a checkbox labeled 'Use Dedicated Folder for Storing Libraries', which is currently unchecked. Below the checkbox is a 'Libraries Folder' text input field and another 'Browse...' button. At the bottom of the right pane, there is a note: 'Different users and projects can share the same compilation libraries (see Help for details).'. A mouse cursor is visible at the bottom right of the dialog box.

Steps

1. Choose Project
- 2. Name and Location**
3. Server and Settings
4. Frameworks

Name and Location

Project Name: Login

Project Location: C:\xampp\tomcat\webapps Browse...

Project Folder: C:\xampp\tomcat\webapps\Login

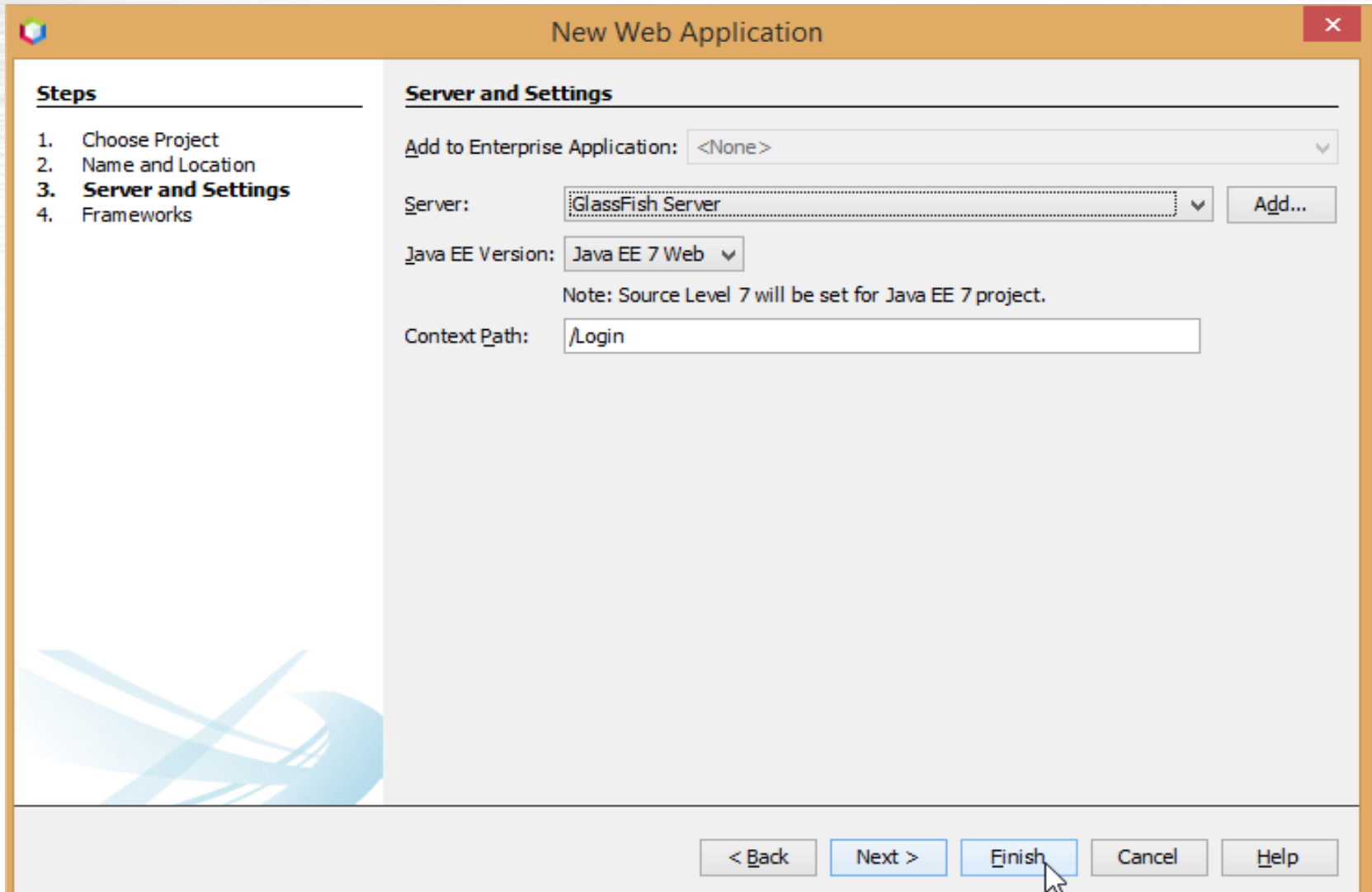
☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

Criando o Projeto no NETBEANS

Server: GlassFish Server



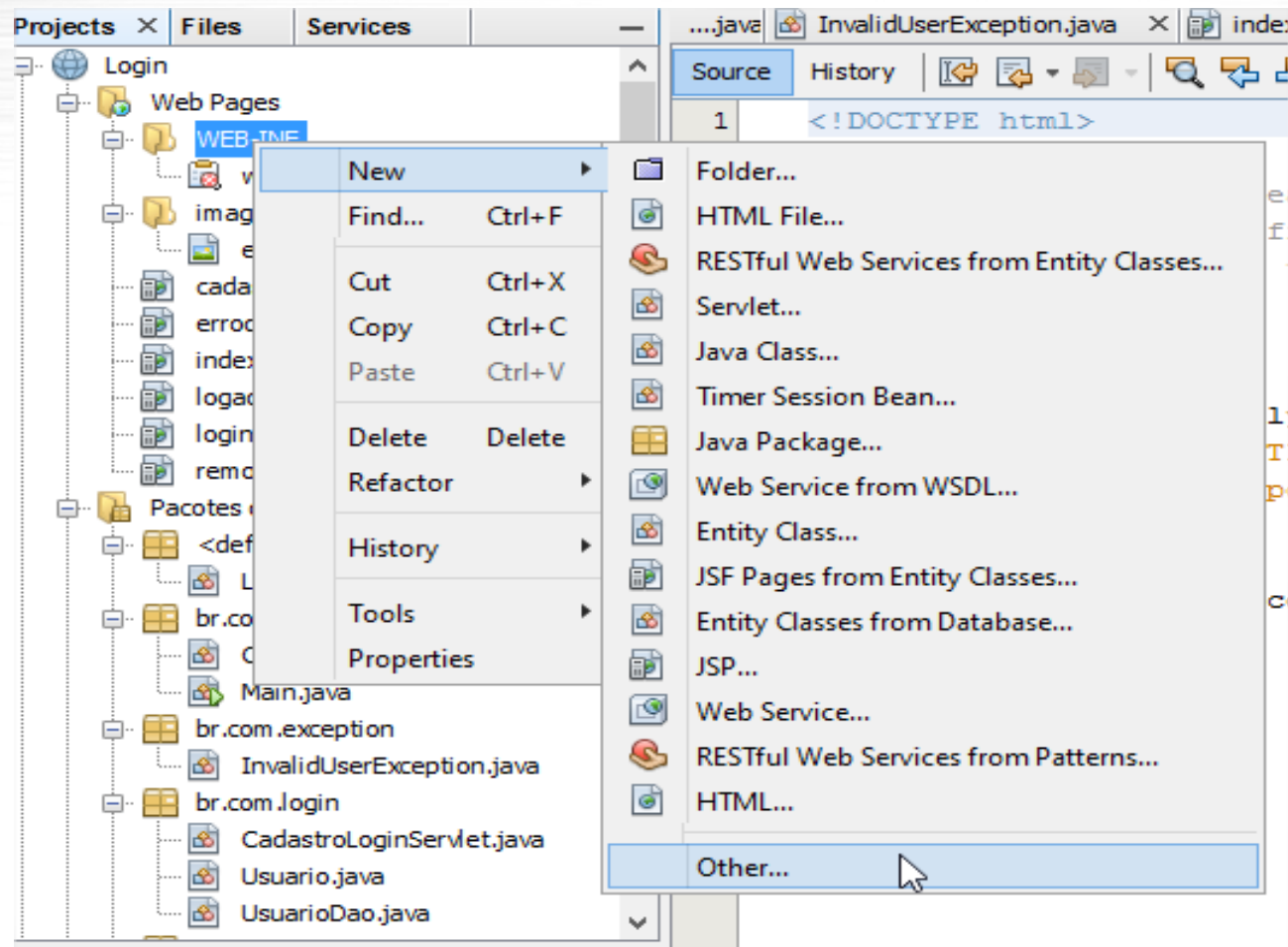
The screenshot shows the 'New Web Application' wizard in NetBeans. The window has a title bar with the NetBeans logo and a close button. The main area is divided into two panes. The left pane, titled 'Steps', contains a list of four steps: 1. Choose Project, 2. Name and Location, 3. **Server and Settings** (which is highlighted), and 4. Frameworks. The right pane, titled 'Server and Settings', contains the following fields and controls:

- 'Add to Enterprise Application:' with a dropdown menu showing '<None>'.
- 'Server:' with a dropdown menu showing 'GlassFish Server' and an 'Add...' button to its right.
- 'Java EE Version:' with a dropdown menu showing 'Java EE 7 Web'.
- A note: 'Note: Source Level 7 will be set for Java EE 7 project.'
- 'Context Path:' with a text input field containing '/Login'.

At the bottom of the wizard, there are five buttons: '< Back', 'Next >', 'Finish' (which is highlighted by a mouse cursor), 'Cancel', and 'Help'.

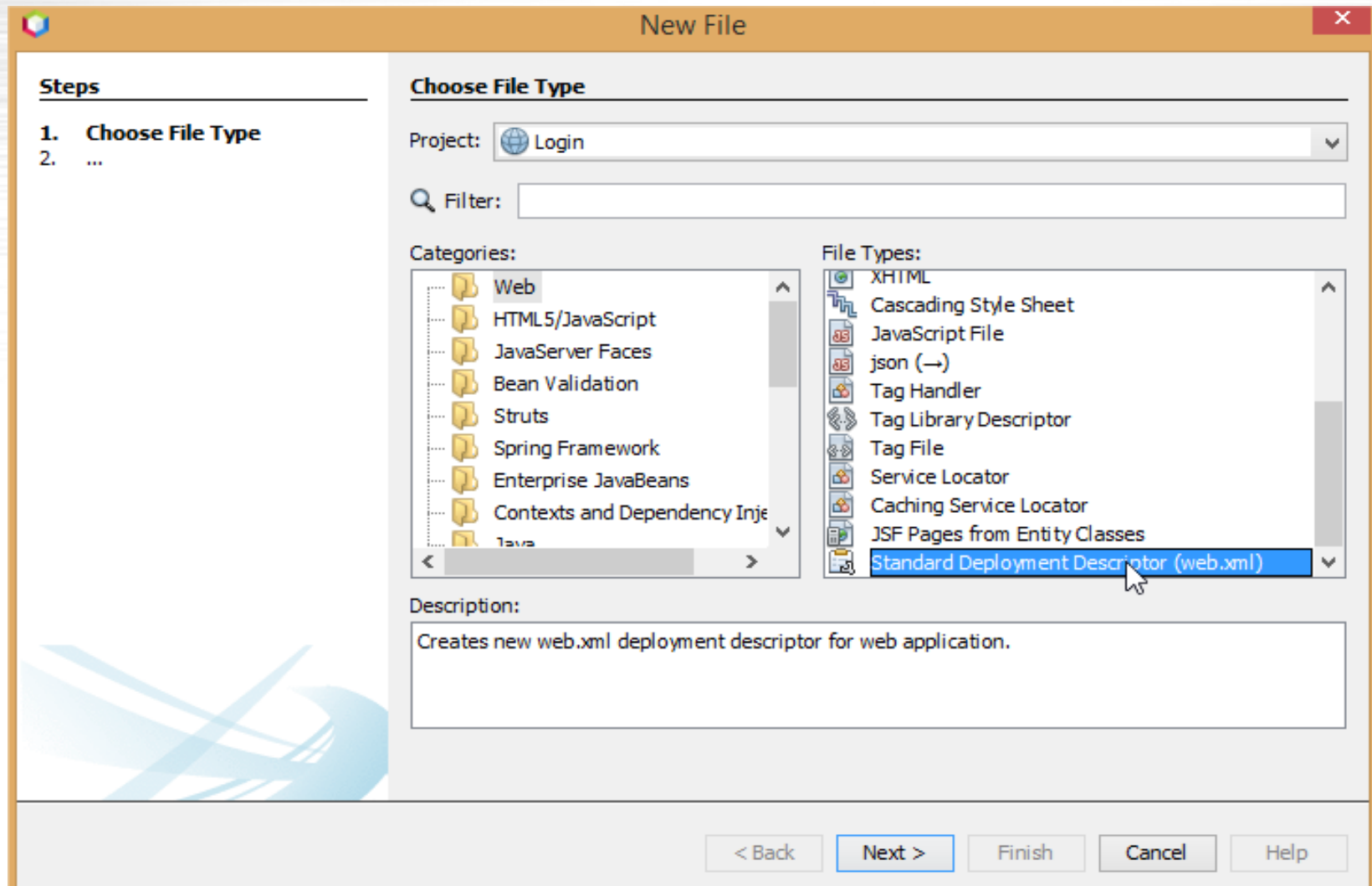
Criando o web.xml

New ... Other ...



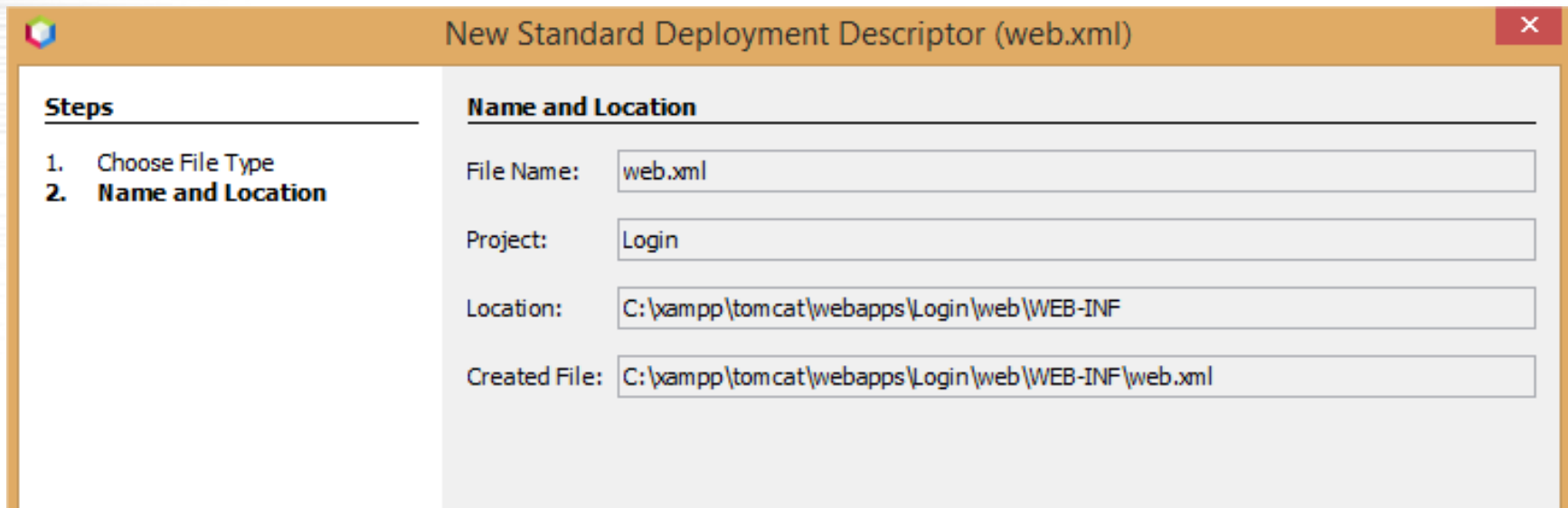
Criando o web.xml

Web ... web.xml



Criando o web.xml

Finish e o arquivo será criado na pasta WEB-INF



New Standard Deployment Descriptor (web.xml)

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

File Name:

Project:

Location:

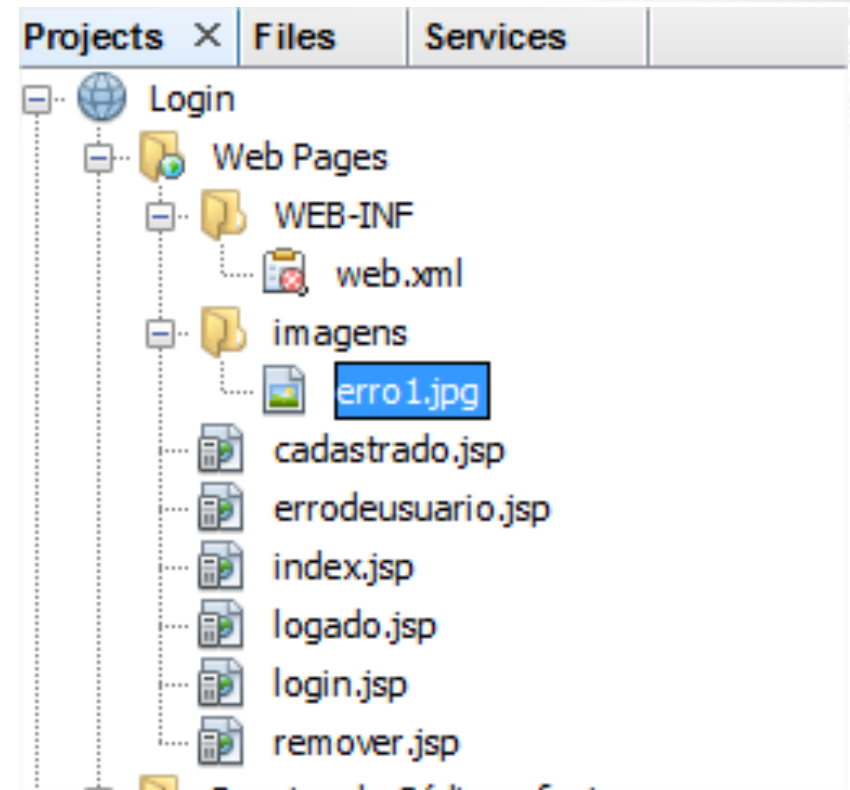
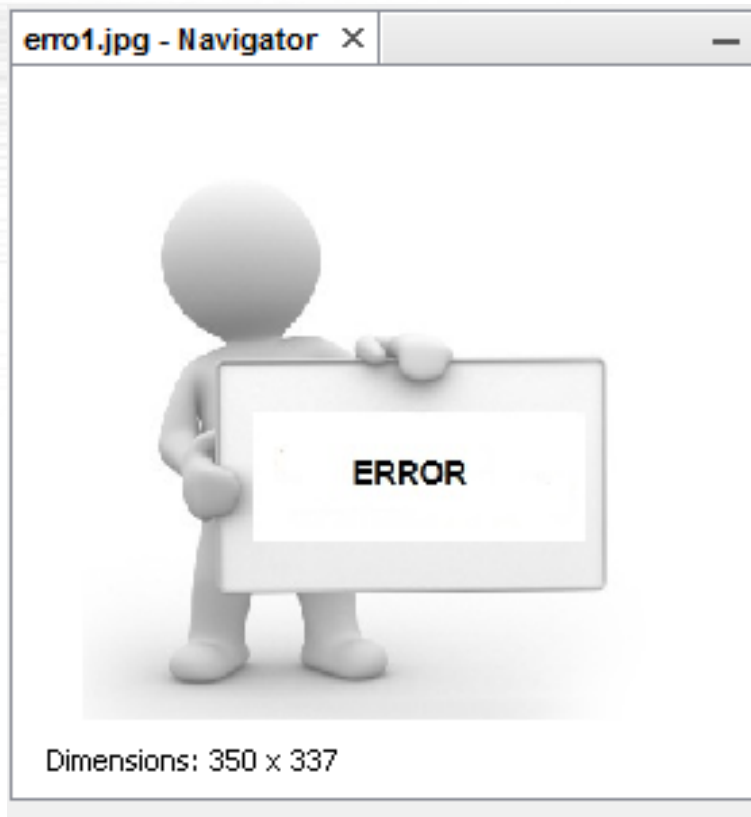
Created File:

Código do web.xml

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
3      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4      xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
5          http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd" >
6      <servlet>
7          <servlet-name>CadastroLoginServlet</servlet-name>
8          <servlet-class>br.com.login.CadastroLoginServlet</servlet-class>
9      </servlet>
10     <servlet>
11         <servlet-name>LoginControllers</servlet-name>
12         <servlet-class>LoginControllers</servlet-class>
13     </servlet>
14     <servlet-mapping>
15         <servlet-name>CadastroLoginServlet</servlet-name>
16         <url-pattern>/CadastroLogin</url-pattern>
17     </servlet-mapping>
18     <servlet-mapping>
19         <servlet-name>LoginControllers</servlet-name>
20         <url-pattern>/Login</url-pattern>
21     </servlet-mapping>
22     <session-config>
23         <session-timeout>
24             30
25         </session-timeout>
26     </session-config>
27 </web-app>
```

Imagem do Projeto (erro1.jpg)

A imagem erro1.jpg (enviar no Blog)



Index.jsp

Source

History



```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
4     "http://www.w3.org/TR/html4/loose.dtd">
5 <html>
6 <head>
7 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
8 <title>Insert title here</title>
9 </head>
10 <body>
11     <h1>Cadastrar Login</h1>
12     <table>
13         <form name="frmCadastrarLogin" action="CadastroLogin" method="post">
14
15             <tr>
16                 <td>Email.:</td><td><input type="text" name="email"></td>
17                 <td>Senha.:</td><td><input type="password" name="senha"></td>
18
19                 <td colspan="2"><input type="submit" value="cadastrar"/></td>
20
21             </form>
22
23         </table>
24 </body>
</html>
```

login.jsp → Parte 1

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<script type="text/javascript">
    function validarLogin(){
        if(document.formLogin.email.value=== "") {
            alert("Campo Usuário Não Informado");
            return false;
        }
        if(document.formLogin.senha.value=== "") {
            alert("Campo Senha Não Informado");
            return false;
        }

        document.formLogin.submit();
    }
</script>
<title>Acessar</title>
</head>
```

login.jsp → Parte 2

```
20 </script>
21 <title>Acessar</title>
22 </head>
23 <body>
24 <!-- <form name="formLogin" action="Login" method="post"> -->
25 <form name="formLogin" action="Login" method="post">
26   <p align="center">
27     <table>
28       <tr><td>Usuário:</td><td><input type="text" name="email"/></td></tr>
29       <tr><td>Senha:</td><td><input type="password" name="senha"/></td></tr>
30       <tr><td colspan="2" align="center"><input type="button" value="Entrar"
31         onclick="validarLogin()" /></td></tr>
32     </table>
33   </p>
34 </form>
35 </body>
36 </html>
```

cadastrado.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<script language="Javascript">
    var timer = 2;
    function countdown() {
        if (timer > 0) {
            timer -= 1;
            setTimeout("countdown()", 3000);
        } else {
            location.href = 'login.jsp';
        }
    }
    countdown();
</script>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    <%      String email = (String) request.getAttribute("email");
    out.println("<h4>Email " + email + " Cadastrado com sucesso<h4>");
    out.println("Você será redirecionado para pagina de login");
    %>
</body>
</html>
```


errodeusuario.jsp → Parte 1

```
<%@page isErrorPage="true" %>
<%@page import="java.util.Enumeration" %>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
    "http://www.w3.org/TR/html4/loose.dtd">

<html>
<head>
<style type="text/css">
    body{ background-color:#fff;}
    #cabecalho{
        background-color:#fff;
        width:800px;
        height:100px;
    }
    #corpo{
        background-color:#fff;
        width:800px;
        height:450px;
    }
    #rodape{
        background-color:#fff;
        width:800px;
        height:100px;
    }
</style>
```

errodeusuario.jsp → Parte 2

```
28
29 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
30 <title>Error</title>
31 </head>
32 <body>
33 <body>
34 <div align="center">
35   <div id="cabecalho">    </div>
36   <div id="corpo">
37
38     
39     <hr/>
40     <h3> Email e Senha incorretos </h3><br/>
41     <br/>
42     <hr/>
43     <a href="login.jsp">Tentar Novamente</a> <a href="index.jsp">Cadastre-se</a>
44
45   </div>
46   <div id="rodape">    </div>
47
48 </div>
49
50 </body>
51 </html>
```

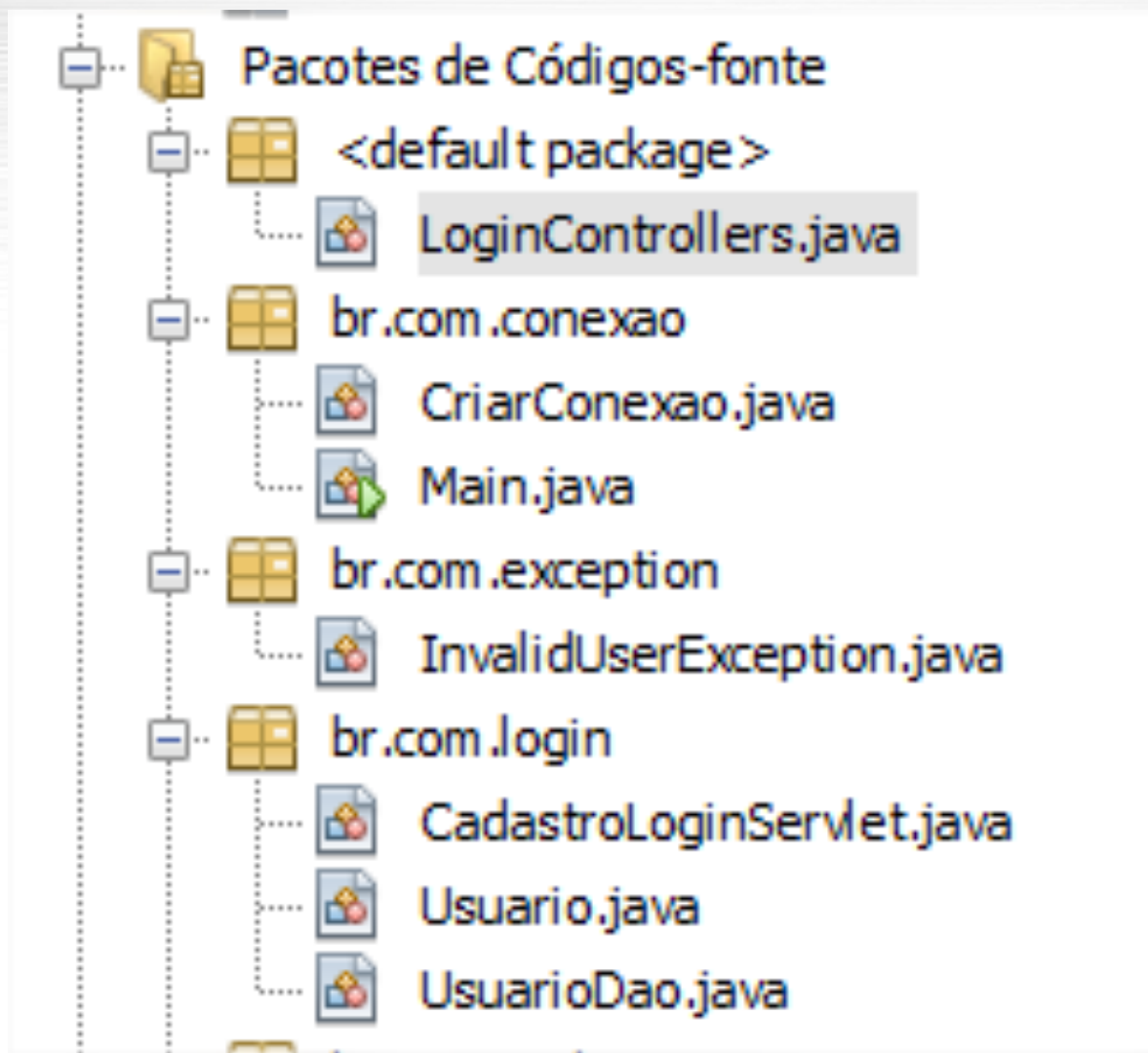
logado.jsp

```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
4     "http://www.w3.org/TR/html4/loose.dtd">
5 <html>
6 <head>
7 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
8 <title>Logado</title>
9 </head>
10 <body>
11     <%
12         String email = (String) session.getAttribute("email");
13         %>
14
15     Seja bem Vindo : <%=email %> | <a href="remover.jsp">Sair</a>
16
17     <h1>Logado</h1>
18     <hr>
19 </body>
20 </html>
```

remover.jsp

```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2     pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
4     "http://www.w3.org/TR/html4/loose.dtd
```

Estrutura dos arquivos .java



LoginControllers.java → Parte 1

```
1  import br.com.conexao.CriarConexao;
2  import java.io.IOException;
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import java.sql.ResultSet;
6  import java.sql.SQLException;
7  import javax.servlet.ServletException;
8  import javax.servlet.http.HttpServlet;
9  import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11 import javax.servlet.http.HttpSession;
12
13 public class LoginControllers extends HttpServlet {
14
15     // private static final long serialVersionUID = 1L;
16
17     @Override
18     protected void doPost(HttpServletRequest request, HttpServletResponse response)
19         throws ServletException, IOException {
20         String emailBuscado = "";
21         String senhaBuscada = "";
22         Connection con;
23         String email = request.getParameter("email");
24         String senha = request.getParameter("senha");
25         String sql = "Select * from tb_login where email = ? and senha = ?";
26     }
```

LoginControllers.java → Parte 2

```
28     try {
29         con = CriarConexao.getConexao();
30         PreparedStatement stmt = con.prepareStatement(sql);
31         stmt.setString(1, email);
32         stmt.setString(2, senha);
33
34         ResultSet rs = stmt.executeQuery();
35         while (rs.next()) {
36             emailBuscado = rs.getString("email");
37             senhaBuscada = rs.getString("senha");
38         }
39         rs.close();
40         stmt.close();
41     } catch (SQLException e) {
42         e.printStackTrace();
43     }
44     if (emailBuscado.equals(email) || senhaBuscada.equals(senha)) {
45         HttpSession session = request.getSession();
46         session.setAttribute("email", email);
47         request.getRequestDispatcher("logado.jsp").forward(request, response);
48     } else {
49         System.out.println(emailBuscado + "-" + email);
50         System.out.println(senhaBuscada + "-" + senha);
51         request.getRequestDispatcher("errodeusuario.jsp").forward(request, response);
52     }
53 }
54 }
```


CriarConexao.java

```
1
2 package br.com.conexao;
3
4 import java.sql.Connection;
5 import java.sql.DriverManager;
6 import java.sql.SQLException;
7
8 public class CriarConexao {
9
10     public static Connection getConexao() throws SQLException {
11         try {
12             Class.forName("com.mysql.jdbc.Driver");
13             System.out.print("conectou");
14             return DriverManager.getConnection("jdbc:mysql://localhost:3306/Login", "root", "root");
15         } catch (SQLException e) {
16             throw new SQLException(e);
17         } catch (ClassNotFoundException e1) {
18             throw new SQLException(e1);
19         }
20     }
21 }
22 }
```

Main.java

```
1
2  package br.com.conexao;
3
4  import java.sql.Connection;
5  import java.sql.SQLException;
6
7
8  public class Main {
9
10     public static void main(String[] args) throws SQLException {
11
12         Connection con = CriarConexao.getConexao();
13     }
14 }
```

InvalidUserException.java

```
1
2  package br.com.exception;
3
4  import javax.servlet.ServletException;
5
6
7  public class InvalidUserException extends ServletException {
8
9      public InvalidUserException(String message) {
10         super(message);
11     }
12
13 }
```

```
package br.com.login;
```

```
- import br.com.conexao.CriarConexao;  
import java.io.IOException;  
import java.io.PrintWriter;  
import java.sql.Connection;  
import java.sql.SQLException;  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServlet;  
import javax.servlet.http.HttpServletRequest;  
import javax.servlet.http.HttpServletResponse;
```

```
public class CadastroLoginServlet extends HttpServlet {
```

```
    private static final long serialVersionUID = 1L;
```

```
-    public CadastroLoginServlet() {  
        super();  
    }
```

CadastroLoginServlet.java Parte 2

```
23 @Override
24 protected void doPost(HttpServletRequest request, HttpServletResponse response)
25     throws ServletException, IOException {
26     PrintWriter writer = response.getWriter();
27     String email = request.getParameter("email");
28     String senha = request.getParameter("senha");
29
30     Connection con;
31     try {
32         con = CriarConexao.getConexao();
33
34         Usuario u = new Usuario();
35         u.setEmail(email);
36         u.setSenha(senha);
37
38         UsuarioDao dao = new UsuarioDao(con);
39         dao.adicionar(u);
40         request.setAttribute("email", u.getEmail());
41         request.getRequestDispatcher("cadastrado.jsp").forward(request, response);
42
43     } catch (SQLException e) {
44         e.printStackTrace();
45     }
46 }
47 }
```

Usuario.java

```
1  package br.com.login;
2
3  public class Usuario {
4      private int id;
5      private String email;
6      private String senha;
7
8      public int getId() {
9          return id;
10     }
11     public void setId(int id) {
12         this.id = id;
13     }
14     public String getEmail() {
15         return email;
16     }
17     public void setEmail(String usuario) {
18         this.email = usuario;
19     }
20     public String getSenha() {
21         return senha;
22     }
23     public void setSenha(String senha) {
24         this.senha = senha;
25     }
26 }
```

UsuarioDAO.java Parte 1

```
1  package br.com.login;
2
3  import java.sql.Connection;
4  import java.sql.PreparedStatement;
5  import java.sql.SQLException;
6
7  public class UsuarioDao {
8      private Connection con;
9      public UsuarioDao(Connection con) {
10         this.con = con;
11     }
12     public void adicionar(Usuario u) throws SQLException {
13
14         String sql = "insert into tb_login(email,senha) values(?,?)";
15
16         try {
17             PreparedStatement stmt = con.prepareStatement(sql);
18
19             stmt.setString(1, u.getEmail());
20             stmt.setString(2, u.getSenha());
21
22             stmt.execute();
23             stmt.close();
24         }
```


UsuarioDAO.java Parte 2

```
24
25         } catch (SQLException e) {
26             e.printStackTrace();
27         } finally {
28             con.close();
29         }
30     }
31
32 }
33
```

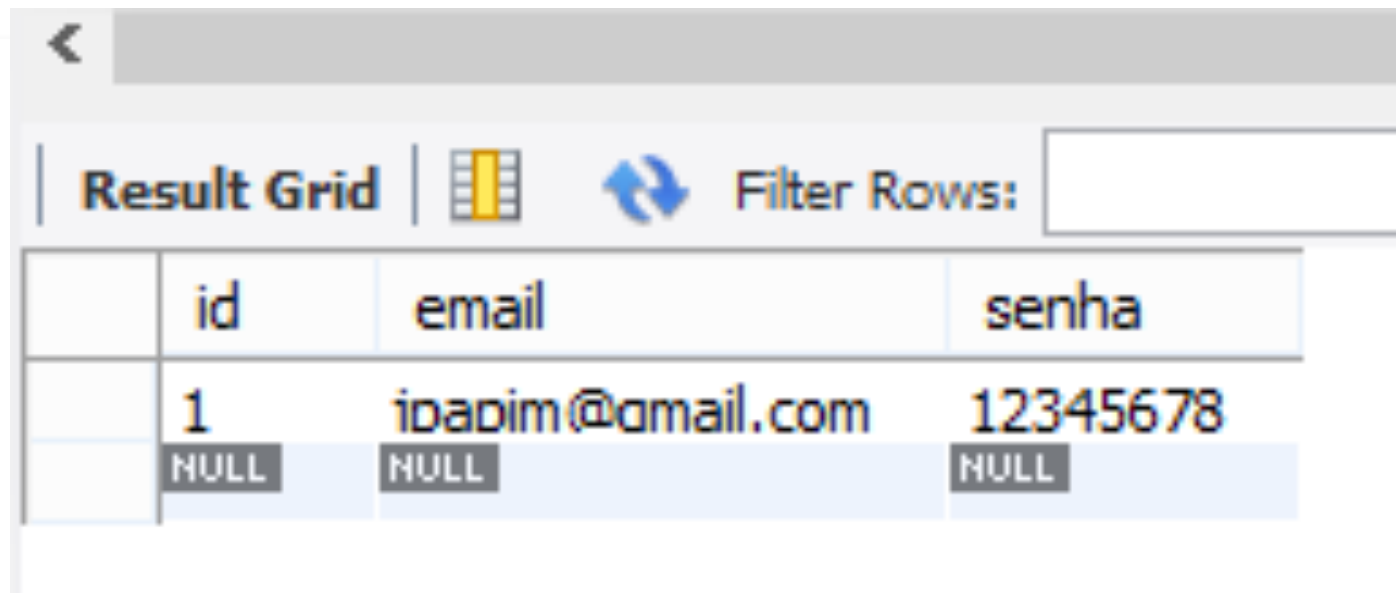
Executando a aplicação

Cadastrar Login

Email.: jpapim@gmail.com

Senha.:

cadastrar



The screenshot shows a web application interface with a 'Result Grid' section. The grid contains a table with columns 'id', 'email', and 'senha'. The first row shows a user with id '1', email 'jpapim@gmail.com', and password '12345678'. Below this row, there are three cells containing the word 'NULL'. The interface also includes a 'Filter Rows' input field and a refresh icon.

| | id | email | senha |
|--|------|------------------|----------|
| | 1 | jpapim@gmail.com | 12345678 |
| | NULL | NULL | NULL |

Executando a aplicação

Usuário: jpapim@gmail.com

Senha:

Entrar



Email e Senha incorretos

[Tentar Novamente](#) [Cadastre-se](#)

Executando a aplicação

Usuário: jpapim@gmail.com

Senha:

Entrar



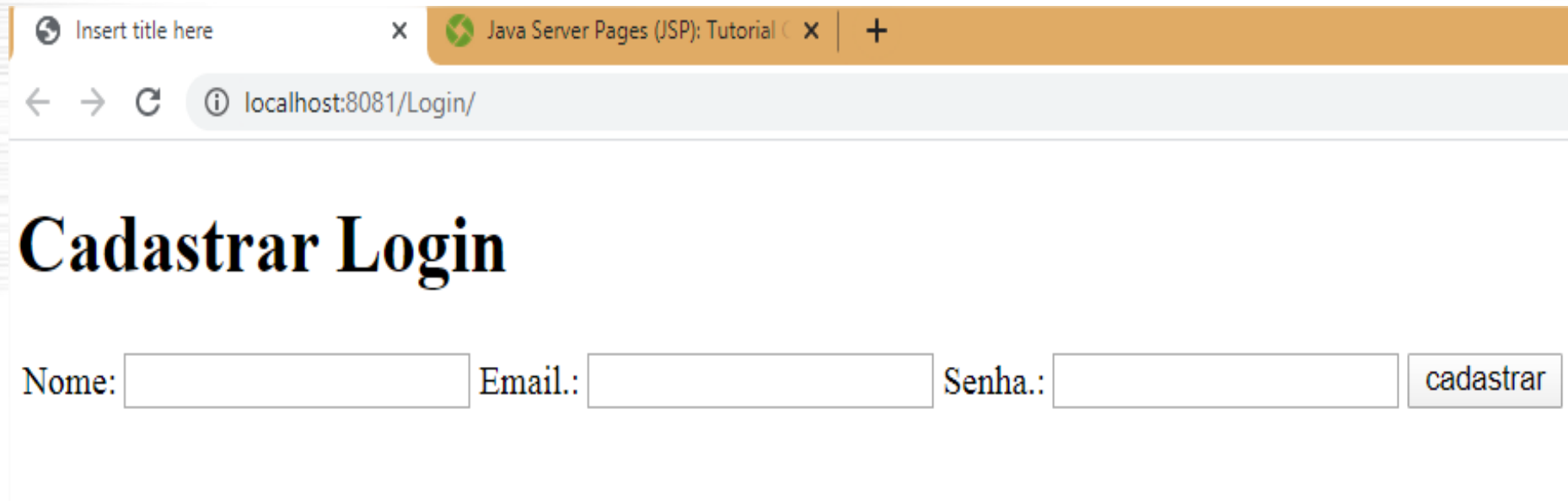
localhost:8081/Login/Login

Seja bem Vindo : jpapim@gmail.com | [Sair](#)

Logado

"Zé-cício"

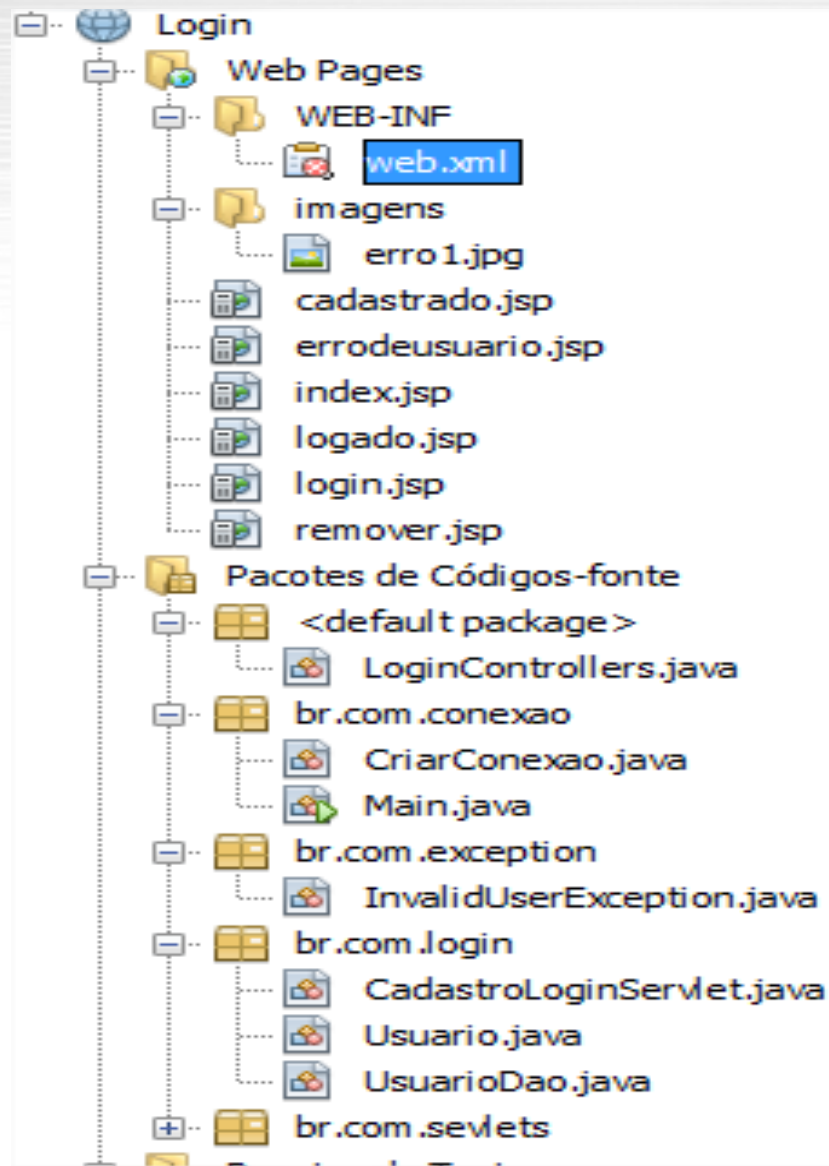
Alterar o Banco de Dados, arquivos .jsp e .java da aplicação de Cadastrar Login, ou seja, incluindo o campo “Nome”, conforme abaixo:



The screenshot shows a web browser window with two tabs: 'Insert title here' and 'Java Server Pages (JSP): Tutorial'. The address bar displays 'localhost:8081/Login/'. The main content area features a heading 'Cadastrar Login' in a large, bold, black serif font. Below the heading is a registration form with three input fields labeled 'Nome:', 'Email.:', and 'Senha.:'. Each label is followed by a text input box. To the right of the 'Senha.' field is a light gray button with the text 'cadastrar' in a dark gray font.

Nome: Email.: Senha.:

Implementar seguindo o modelo apresentado na aula:





Por hoje é só !!!

Até a próxima aula...
