



Estácio

Universidade Estácio de Sá

- DESENVOLVIMENTO FULL STACK- TURMA 23.3 -9003
 - Disciplina: RPG0017 - Vamos Integrar Sistemas.
 - Semestre Letivo: 2023.2
 - Repositorio Git: <https://github.com/Gregdev22/Missao-4-Mundo-3>
-

- [EMERSON GREGORIO ALVES](#) - MATRICULA: 2022.0908.4986
-

Missão Prática | Nível 4 | Mundo 3

Implementação de sistema cadastral com interface Web, baseado nas tecnologias deServlets, JPA e JEE.

Procedimento 1: Camadas de Persistência e Controle

Procedimento 2: Interface Cadastral com Servlet e JSPs

Procedimento 3: Melhorando o Design da Interface



Objetivos da Prática

- Implementar persistência com base em JPA.

- Implementar regras de negócio na plataforma JEE, através de EJBs.
 - Implementar sistema cadastral Web com base em Servlets e JSPs.
 - Utilizar a biblioteca Bootstrap para melhoria do design.
 - No final do exercício, o aluno terá criado todos os elementos necessários para exibição e entrada de dados na plataforma Java Web, tornando-se capacitado para lidar com contextos reais de aplicação.
-

Códigos

Procedimento 1: Camadas de Persistência e Controle

- ServletProduto.java

```

/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
 default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to
 edit this template
 */
package cadastroee.servlets;

import cadastroee.controller.ProdutoFacadeLocal;
import jakarta.ejb.EJB;
import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import cadastroee.model.Produto;

/**
 *
 * @author grego
 */
@WebServlet("/ServletProduto")
public class ServletProduto extends HttpServlet {

    @EJB
    ProdutoFacadeLocal facade;

    /**
     * Processes requests for both HTTP <code>GET</code> and
     <code>POST</code>
     * methods.
     *
     * @param request servlet request
     * @param response servlet response

```

```

    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */

    protected void processRequest(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following sample code.
            */

            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet ServletProduto</title>");
            out.println("</head>");
            out.println("<body>");
            // out.println("<h1>Servlet ServletProduto at " +
request.getContextPath() + "</h1>");
            //out.println(facade.findAll().getClass());
            //out.println(facade.find(1).getClass());

            for (Produto p : facade.findAll()) {
                out.println("<li> " + p.getNome() + "</li>");
            }

            out.println("</body>");
            out.println("</html>");
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServletRequest methods. Click
on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse
response)

```

```

        throws ServletException, IOException {
    processRequest(request, response);
}

/**
 * Returns a short description of the servlet.
 *
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
} // </editor-fold>
}

```

Procedimento 2: Interface Cadastral com Servlet e JSPs

- ServletProduto.java

```

/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
 * default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to
 * edit this template
 */
package cadastroee.servlets;

import cadastroee.controller.ProdutoFacadeLocal;
import jakarta.ejb.EJB;
import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import cadastroee.model.Produto;

/**
 *
 * @author grego
 */
@WebServlet("/ServletProduto")
public class ServletProduto extends HttpServlet {

    @EJB
    ProdutoFacadeLocal facade;

    /**
     * Processes requests for both HTTP <code>GET</code> and
     * <code>POST</code>
     * methods.
     *

```

```

    * @param request servlet request
    * @param response servlet response
    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */

    protected void processRequest(HttpServletRequest request,
    HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            /* TODO output your page here. You may use following sample code.
            */

            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Servlet ServletProduto</title>");
            out.println("</head>");
            out.println("<body>");
            // out.println("<h1>Servlet ServletProduto at " +
request.getContextPath() + "</h1>");
            //out.println(facade.findAll().getClass());
            //out.println(facade.find(1).getClass());

            for (Produto p : facade.findAll()) {
                out.println("<li> " + p.getNome() + "</li>");
            }

            out.println("</body>");
            out.println("</html>");
        }
    }

    // <editor-fold defaultstate="collapsed" desc="HttpServletRequest methods. Click
on the + sign on the left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse
response)
        throws ServletException, IOException {
        processRequest(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override

```

```

        protected void doPost(HttpServletRequest request, HttpServletResponse
response)
            throws ServletException, IOException {
            processRequest(request, response);
        }

/**
 * Returns a short description of the servlet.
 *
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
} // </editor-fold>

}

```

- ServletProdutoFC.java

```

/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/JSP_Servlet/Servlet.java to
edit this template
 */
package cadastroee.servlets;

import cadastroee.controller.ProdutoFacadeLocal;
import cadastroee.model.Produto;
import jakarta.ejb.EJB;
import jakarta.servlet.RequestDispatcher;
import jakarta.servlet.annotation.WebServlet;
import java.io.IOException;
import java.io.PrintWriter;
import jakarta.servlet.ServletException;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;

/**
 *
 * @author grego
 */
@WebServlet(name="ServletProdutoFC", urlPatterns = {"/ServletProdutoFC"})
public class ServletProdutoFC extends HttpServlet {

    @EJB
    ProdutoFacadeLocal facade;

    int idAtual =6;

    public int aleatorio() {
        // Math.random() gera um número aleatório entre 0.0 e 0.999
        // Assim, Math.random()*5 estará entre 0.0 e 4.999
        double doubleRandomNumber = Math.random() * 100;
        int randomNumber = (int)doubleRandomNumber;
        return randomNumber;
    }
}

```

```

    }

    /**
     * Processes requests for both HTTP GET and
     * POST
     * methods.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    protected void processRequest(HttpServletRequest request,
        HttpServletResponse response)
        throws ServletException, IOException {
        String acao = request.getParameter("acao");
        String destino = "";

        if(acao != null){
            switch(acao){

                case "listar":
                    request.setAttribute("lista", facade.findAll());
                    destino = "ProdutoLista.jsp";
                    break;

                case "excluir":
                    int idProduto =
Integer.valueOf(request.getParameter("idproduto"));
                    facade.remove(facade.find(idProduto));
                    request.setAttribute("lista", facade.findAll());
                    RequestDispatcher rd =
request.getRequestDispatcher("ProdutoLista.jsp");
                    rd.forward(request, response);
                    break;

                case "formIncluir":
                    destino = "ProdutoDados.jsp";
                    break;

                case "formAlterar":
                    int id_produto =
Integer.valueOf(request.getParameter("idproduto"));
                    request.setAttribute("lista", facade.find(id_produto));
                    destino = "ProdutoDados.jsp";
                    break;
            }

            RequestDispatcher rd =
request.getRequestDispatcher(destino);
            rd.forward(request, response);

        } else{
            request.setAttribute("lista", facade.findAll());
            RequestDispatcher rd =
request.getRequestDispatcher("ProdutoLista.jsp");
            rd.forward(request, response);
        }
    }
}

```

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

```
/**
 * Handles the HTTP <code>GET</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {
    processRequest(request, response);
}

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse
response)
    throws ServletException, IOException {

    String acao = request.getParameter("acao");

    switch(acao){

        case("alterar"):
            int produtoID =
Integer.valueOf(request.getParameter("idproduto"));
            String nome = request.getParameter("nome");
            int quantidadeProduto =
Integer.valueOf(request.getParameter("quantidade"));
            float precoProduto =
Float.valueOf(request.getParameter("preco"));
            Produto produtoAlterar = facade.find(produtoID);

produtoAlterar.setNome(request.getParameter("nome"));
            produtoAlterar.setQuantidade(quantidadeProduto);
            produtoAlterar.setPrecoVenda(precoProduto);
            facade.edit(produtoAlterar);
            request.setAttribute("lista", facade.findAll());
            RequestDispatcher rd =
request.getRequestDispatcher("ProdutoLista.jsp");
            rd.forward(request, response);
            break;

        case("incluir"):
            int idNext = aleatorio();
```



```

        if (idNext != idAtual){
            float preco =
Float.valueOf(request.getParameter("preco"));
            String nome2 = request.getParameter("nome");
            int quantidade =
Integer.valueOf(request.getParameter("quantidade"));
            Produto produto = new Produto(idNext,
                request.getParameter("nome"),
                quantidade,
                preco);
            facade.create(produto);
            request.setAttribute("lista",
facade.findAll());

            idAtual = idNext;
            RequestDispatcher rd2 =
request.getRequestDispatcher("ProdutoLista.jsp");
            rd2.forward(request, response);
            break;
        } else {
            idNext = idNext++;
            idAtual = idNext;
            float preco =
Float.valueOf(request.getParameter("preco"));
            String nome2 = request.getParameter("nome");
            int quantidade =
Integer.valueOf(request.getParameter("quantidade"));
            Produto produto = new Produto(idNext,
                request.getParameter("nome"),
                quantidade,
                preco);
            facade.create(produto);
            request.setAttribute("lista",
facade.findAll());

            idAtual = idNext;
            RequestDispatcher rd2 =
request.getRequestDispatcher("ProdutoLista.jsp");
            rd2.forward(request, response);
            break;
        }
    }

}

/**
 * Returns a short description of the servlet.
 *
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
} // </editor-fold>

}

```

- ProdutoLista.jsp

```

<%--
Document    : ProdutoLista
Created on  : 3 de out. de 2023, 21:16:09

```

Author : grego
--%>

```
<%@page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>
```

```
<%@page import="cadastroee.model.Produto"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@page import="cadastroee.controller.ProdutoFacadeLocal"%>
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>
    <div>
      <h1>Listagem de Produtos</h1>
      <h3>
        <a href="ServletProdutoFC?acao=formIncluir"> Novo Produto
      </a>
      <!-- <a href="ServletProdutoFC"> Atualizar Dados </a> !-->
    </h3>

    <table border="1" width="100%">
      <tr>
        <td> ID </td>
        <td> Nome </td>
        <td> Quantidade </td>
        <td> Preço de Venda </td>
        <td> Opções </td>
      </tr>

      <%
        try{
          List<Produto> lista = (List)
request.getAttribute("lista");
          for(Produto p: lista){
            %>
            <tr>
              <td>
                <%=p.getIdproduto()%>
              </td>
              <td>
                <%=p.getNome()%>
              </td>
              <td>
                <%=p.getQuantidade()%>
              </td>
              <td>
                <%=p.getPrecoVenda()%>
              </td>
              <td>
```

```

                                <a
href="ServletProdutoFC?acao=formAlterar&idproduto=<%=p.getIdproduto()%>">
Alterar </a>
                                <a
href="ServletProdutoFC?acao=excluir&idproduto=<%=p.getIdproduto()%>"> Excluir
</a>
                                </td>

                                </tr>
                                <% }
                                } catch(NullPointerException nexc){
out.print("<h1>" + nexc.getMessage() + "</h1>");

                                }
                                %>

                                </table>
                                </div>
                                </body>
</html>

```

- ProdutoDados.jsp

```

<%--
Document    : ProdutoDados
Created on  : 5 de out. de 2023, 14:26:07
Author      : grego
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="cadastroee.model.Produto"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@page import="cadastroee.controller.ProdutoFacadeLocal"%>

<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <title>JSP Page</title>
  </head>
  <body>

    <%
      try{
        Produto produto = (Produto) request.getAttribute("lista");
        if (produto != null){

          <h1> Dados do Produto </h1>
          <form action="ServletProdutoFC" method="post" >
            <input type="hidden" name="acao" value="alterar">
            <input type="hidden" name="idproduto"
value="<%=produto.getIdproduto()%>">
            Nome: <input name="nome" value="<%=produto.getNome()%>" />

```

```

        Quantidade: <input name="quantidade"
value="<%=produto.getQuantidade()%>" />
        Preço de Venda: <input name="preco"
value="<%=produto.getPrecoVenda()%>" />
        <input type="submit" value="Alterar Produto" />

    </form>

<%
    } else {
%>
    <h1> Dados do Produto </h1>
    <form action="ServletProdutoFC" method="post" >
    <input type="hidden" name="acao" value="incluir">
    Nome: <input name="nome" />
    Quantidade: <input name="quantidade" />
    Preço de Venda: <input name="preco" />
    <input type="submit" value="Adicionar Produto" />
    </form>

<%
    }
    } catch(ClassCastException nexc){
        out.print("<h1>"+nexc.getMessage()+"</h1>");}
%>
</body>
</html>

```

[Procedimento 3: Melhorando o Design da Interface](#)

- ProdutoLista.jsp

```

<!--
Document    : ProdutoLista
Created on  : 3 de out. de 2023, 21:16:09
Author      : grego
-->

<%@page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8"%>

<%@page import="cadastroee.model.Produto"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@page import="cadastroee.controller.ProdutoFacadeLocal"%>

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css
" rel="stylesheet" integrity="sha384-

```

```

T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN"
crossorigin="anonymous"> <title>JSP Page</title>
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.mi
n.js" integrity="sha384-
C6RzsynM9kWDrMNeT87bh95OGNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL"
crossorigin="anonymous"></script>
    </head>

    <body class="container">
        <div>
            <h1>Listagem de Produtos</h1>
            <h3>
                <a class="btn btn-primary m-2"
href="ServletProdutoFC?acao=formIncluir"> Novo Produto </a>
                <!-- <a href="ServletProdutoFC"> Atualizar Dados </a> !-->
            </h3>

            <table class="table table-striped" border="1" width="100%">
                <tr class="table-dark">
                    <td> ID </td>
                    <td> Nome </td>
                    <td> Quantidade </td>
                    <td> Preço de Venda </td>
                    <td> Opções </td>
                </tr>

                <%
                    try{
                        List<Produto> lista = (List)
request.getAttribute("lista");
                        for(Produto p: lista){
                            <%>
                            <tr>
                                <td>
                                    <%=p.getIdproduto()%>
                                </td>
                                <td>
                                    <%=p.getNome()%>
                                </td>
                                <td>
                                    <%=p.getQuantidade()%>
                                </td>
                                <td>
                                    <%=p.getPrecoVenda()%>
                                </td>
                                <td>
                                    <a class="btn btn-primary btn-sm"
href="ServletProdutoFC?acao=formAlterar&idproduto=<%=p.getIdproduto()%>">
Alterar </a>
                                    <a class="btn btn-danger btn-sm"
href="ServletProdutoFC?acao=excluir&idproduto=<%=p.getIdproduto()%>"> Excluir
</a>
                                </td>
                            </tr>
                            <% }
                                } catch(NullPointerException nexc){
                                    out.print("<h1>"+nexc.getMessage()+"</h1>");
                                }
                <%>
            </tr>
        <% }
    </body>
    </html>

```

```

        }
    }%>

</table>
</div>
</body>
</html>

```

- ProdutoDados.jsp

```

<!--
    Document    : ProdutoDados
    Created on  : 5 de out. de 2023, 14:26:07
    Author      : grego
-->

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<%@page import="cadastroee.model.Produto"%>
<%@page import="java.util.ArrayList"%>
<%@page import="java.util.List"%>
<%@page import="cadastroee.controller.ProdutoFacadeLocal"%>

<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css
" rel="stylesheet" integrity="sha384-
T3c6CoIi6uLrA9TneNEoa7RxnatzjcDSCmG1MXxSR1GAsXEV/Dwwykc2MPK8M2HN"
crossorigin="anonymous">
        <title>JSP Page</title>
        <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.mi
n.js" integrity="sha384-
C6RzsynM9kWDrmNeT87bh950GNyZPhcTNXj1NW7RuBCsyN/o0jlpcV8Qyq46cDfL"
crossorigin="anonymous"></script>
    </head>
    <body class="container">

        <%
            try{
                Produto produto = (Produto) request.getAttribute("lista");
                if (produto != null){
                    <h1> Dados do Produto </h1>
                    <form class="form" action="ServletProdutoFC" method="post" >
                        <input type="hidden" name="acao" value="alterar">
                        <input type="hidden" name="idproduto"
value="<%=produto.getIdproduto()%>">
                        <div class=" mb-3">
                            <label class="form-label"> Nome: </label>
                            <input class="form-control" name="nome"
value="<%=produto.getNome()%>" />
                        </div>

```

```

        <div class=" mb-3">
            <label class="form-label">Quantidade: </label>
            <input class="form-control" name="quantidade"
value="<%=produto.getQuantidade()%>" />
        </div>
        <div class=" mb-3">
            <label class="form-label"> Preco de Venda: </label>
            <input class="form-control" name="preco"
value="<%=produto.getPrecoVenda()%>" />
        </div>
        <div class=" mb-3">
            <input class=" btn btn-primary" type="submit"
value="Alterar Produto" />
        </div>

    </form>

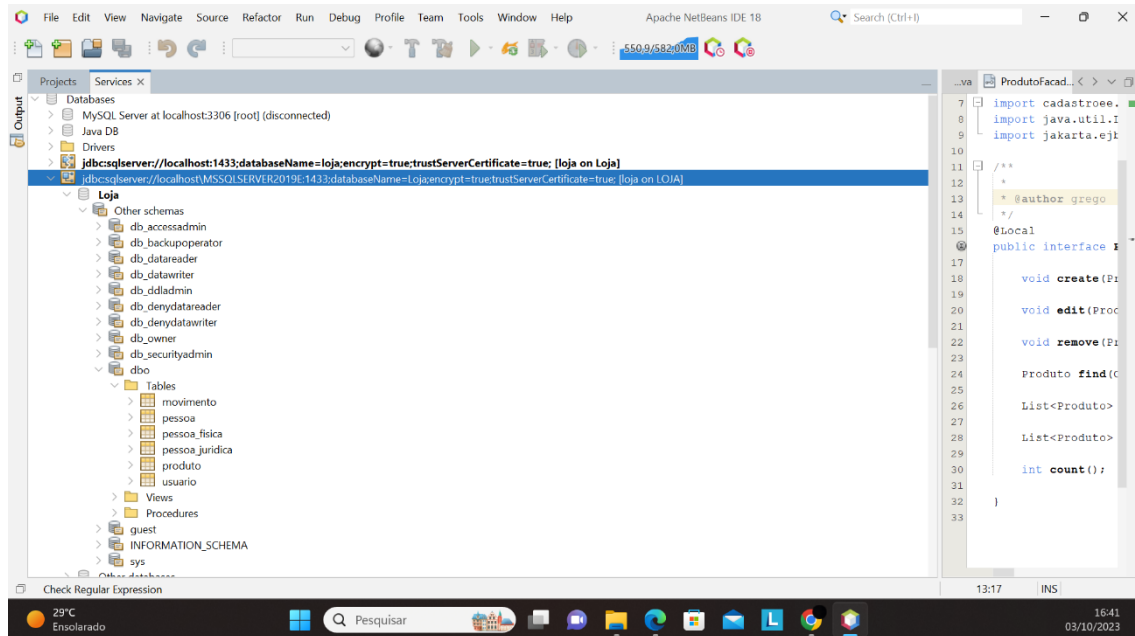
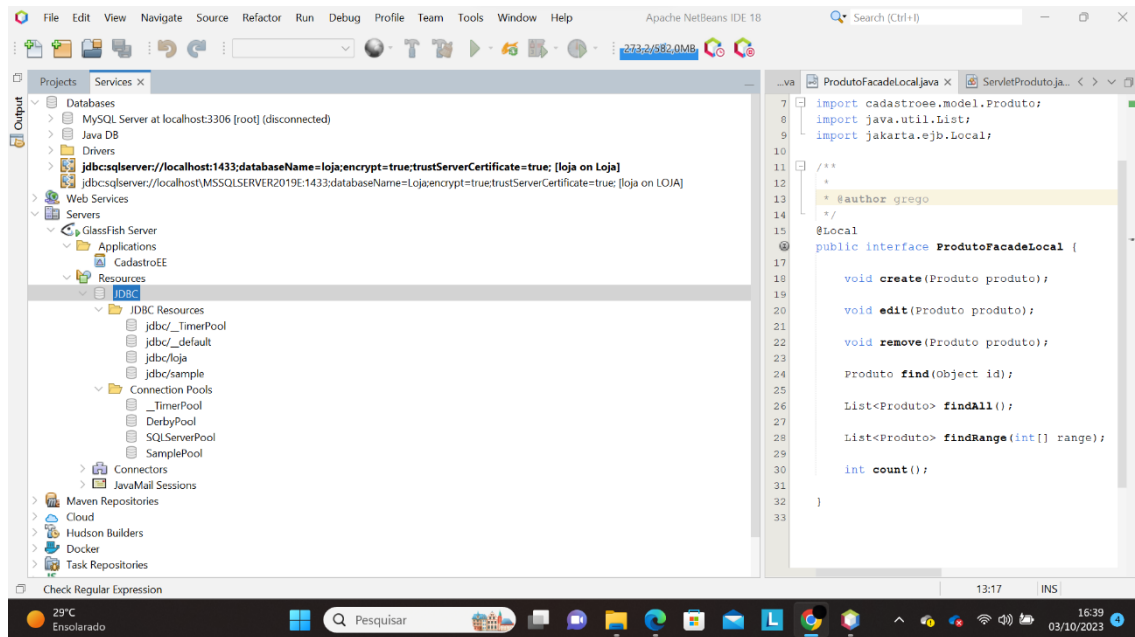
<%
    } else {
%>
    <h1> Dados do Produto </h1>
    <form class="form" action="ServletProdutoFC" method="post" >
        <input type="hidden" name="acao" value="incluir">
        <div class=" mb-3">
            <label class="form-label">Nome: </label>
            <input class="form-control" name="nome" />
        </div>
        <div class=" mb-3">
            <label class="form-label">Quantidade: </label>
            <input class="form-control" name="quantidade" />
        </div>
        <div class=" mb-3">
            <label class="form-label">Preco de Venda: </label>
            <input class="form-control" name="preco" />
        </div>
        <input class=" btn btn-primary" type="submit"
value="Adicionar Produto" />
    </form>

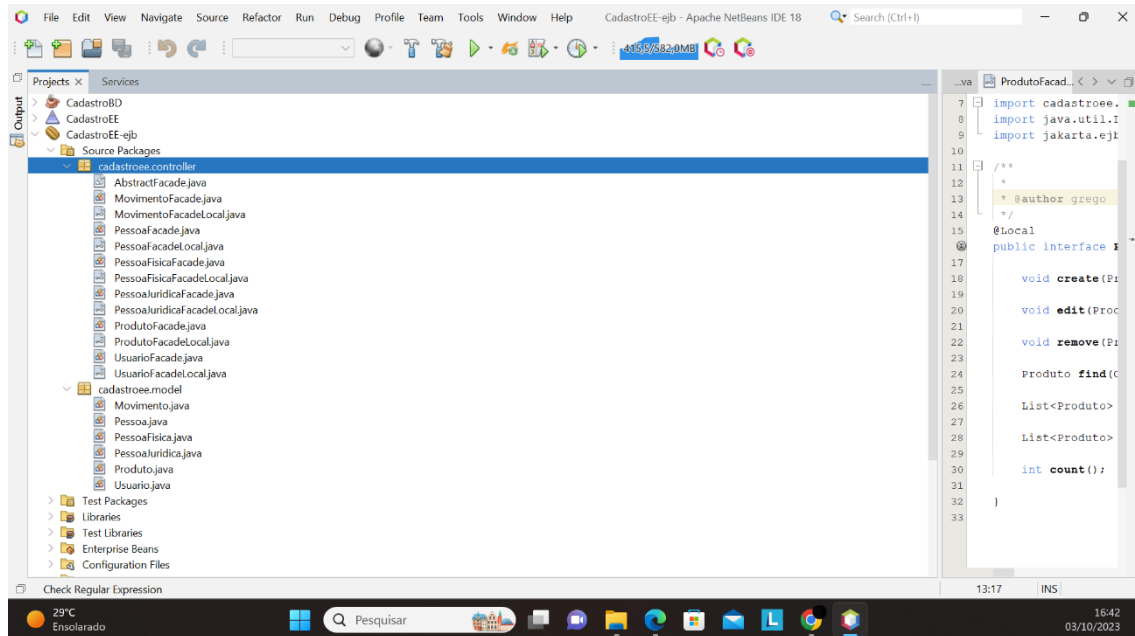
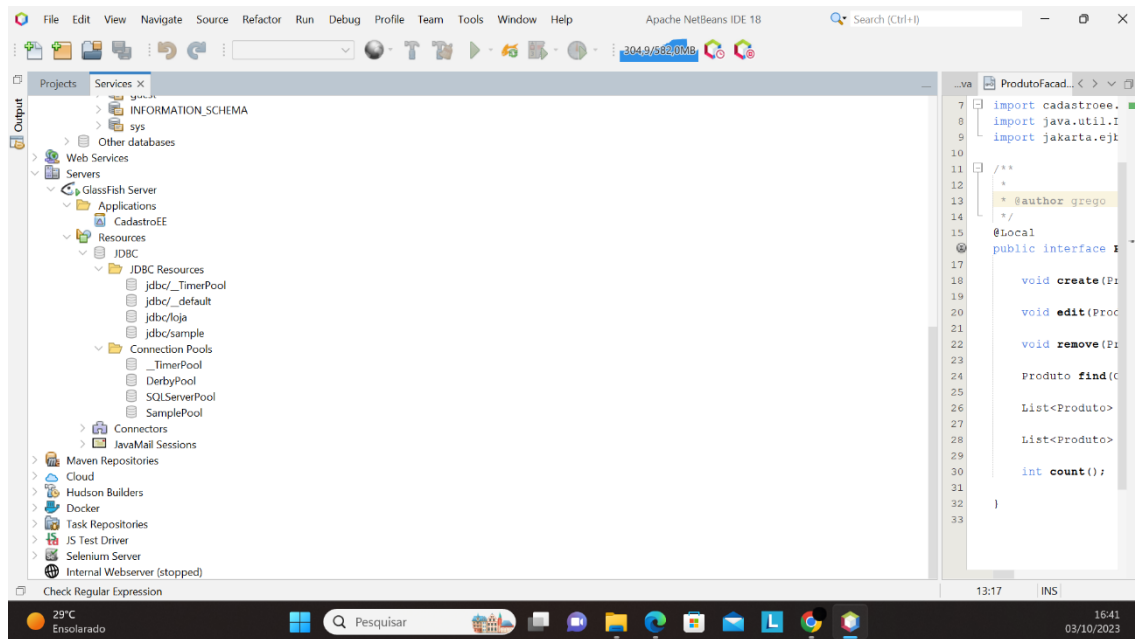
    <%
        }
        } catch(ClassCastException nexc){
            out.print("<h1>" + nexc.getMessage() + "</h1>");
        }
    %>
</body>
</html>

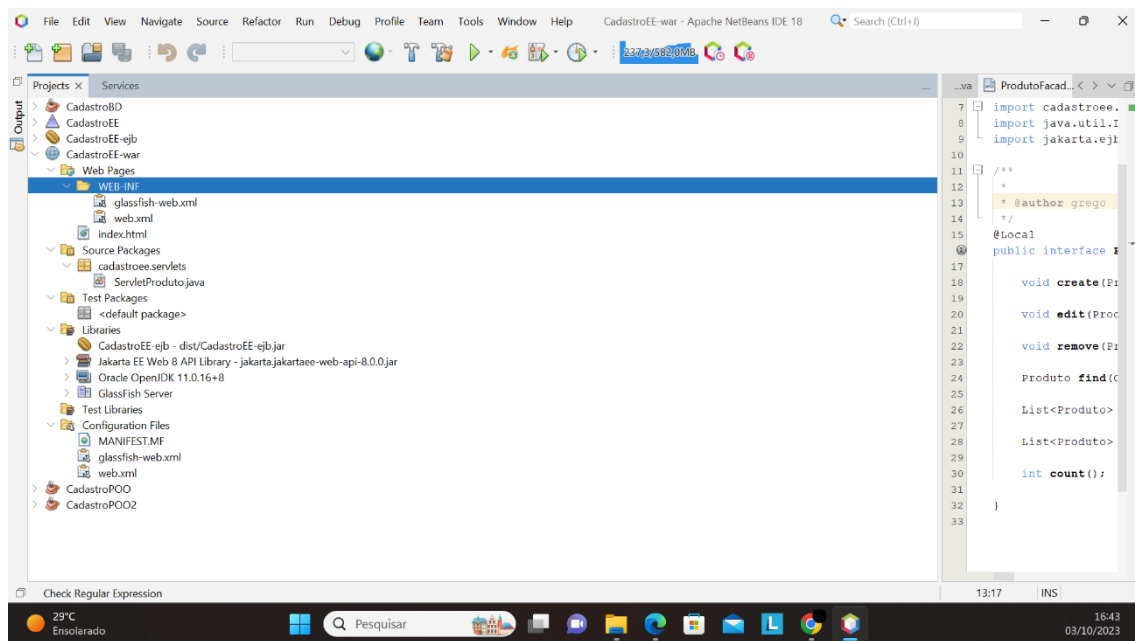
```

Resultados:

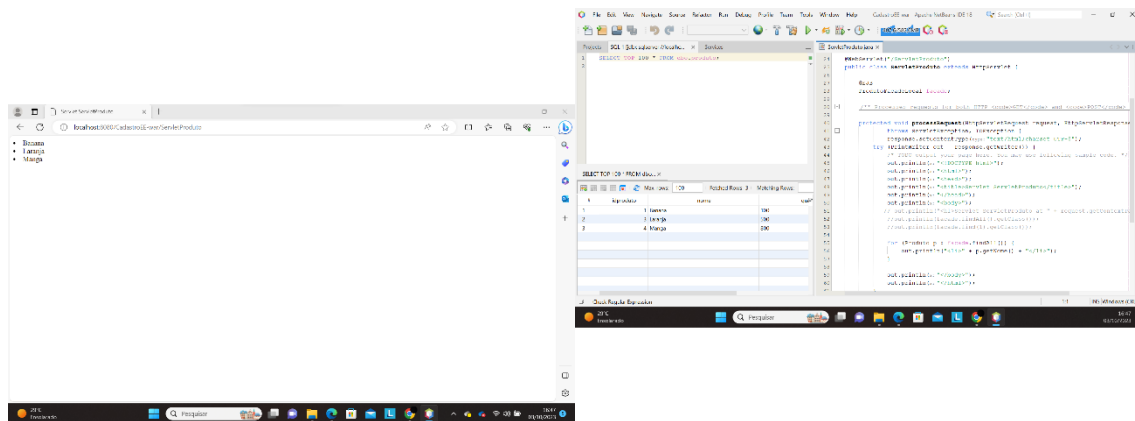
► Procedimento 1: <https://github.com/Gregdev22/Missao-4-Mundo-3/tree/main/Procedimento%201>





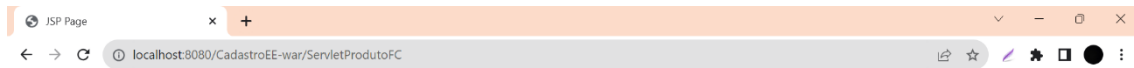


Teste do Servlet:
<http://localhost:8080/CadastroEE-war/ServletProduto>



► Procedimento 2: <https://github.com/Gregdev22/Missao-4-Mundo-3/tree/main/Procedimento%202>

Listar os produtos com a chamada para o endereço seguinte:
<http://localhost:8080/CadastroEE-war/ServletProdutoFC?acao=listar>



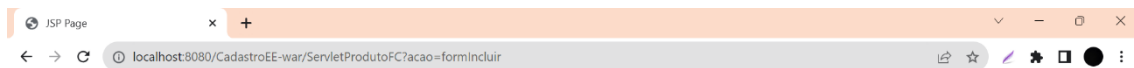
Listagem de Produtos

[Novo Produto](#)

ID	Nome	Quantidade	Preço de Venda	Opções
1	Banana	100	5.0	Alterar Excluir
3	Laranja	500	2.0	Alterar Excluir
4	Manga	800	4.0	Alterar Excluir
14	Pinha	50	50.0	Alterar Excluir

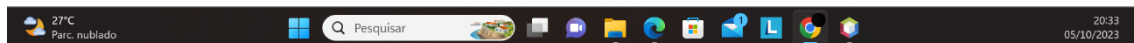


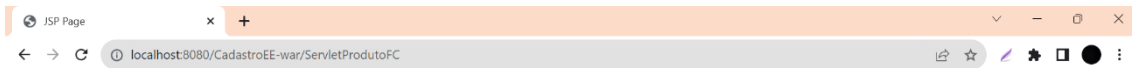
Efetuar uma inclusão a partir do link da tela de listagem



Dados do Produto

Nome: Quantidade: Preço de Venda:

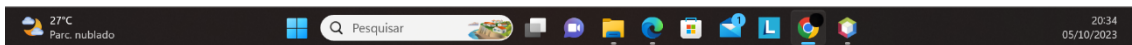




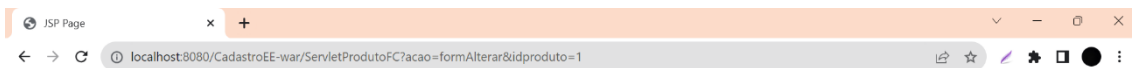
Listagem de Produtos

[Novo Produto](#)

ID	Nome	Quantidade	Preço de Venda	Opções
1	Banana	100	5.0	Alterar Excluir
3	Laranja	500	2.0	Alterar Excluir
4	Manga	800	4.0	Alterar Excluir
5	Pera	150	0.7	Alterar Excluir
14	Pinha	50	50.0	Alterar Excluir



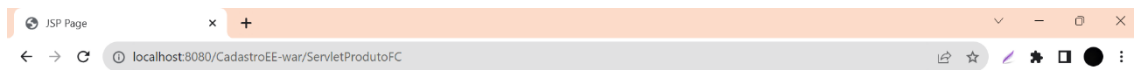
Efetuar uma alteração a partir do link dinâmico da listagem



Dados do Produto

Nome: Quantidade: Preço de Venda:





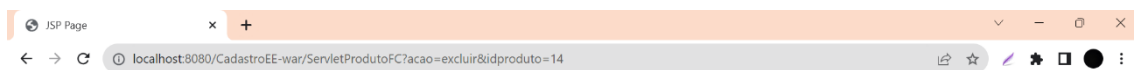
Listagem de Produtos

[Novo Produto](#)

ID	Nome	Quantidade	Preço de Venda	Opções
1	Banana Pacovan	100	5.0	Alterar Excluir
3	Laranja	500	2.0	Alterar Excluir
4	Manga	800	4.0	Alterar Excluir
5	Pera	150	0.7	Alterar Excluir
14	Pinha	50	50.0	Alterar Excluir



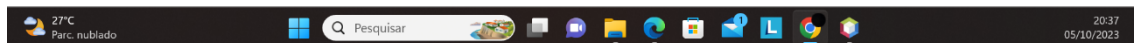
Efetuar uma exclusão a partir do link dinâmico da listagem



Listagem de Produtos

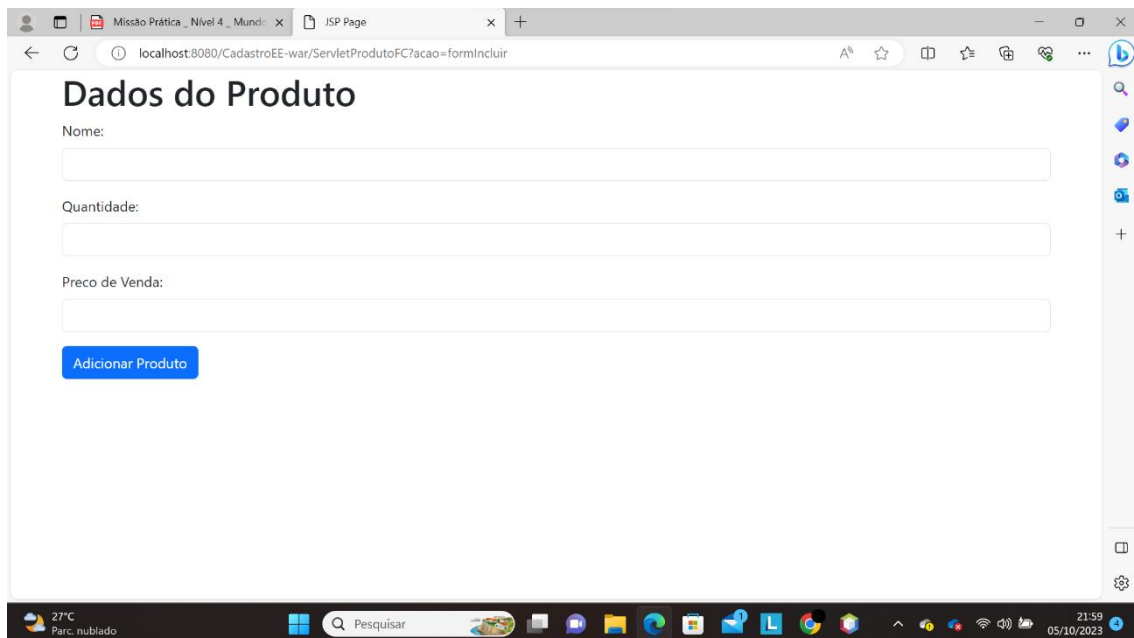
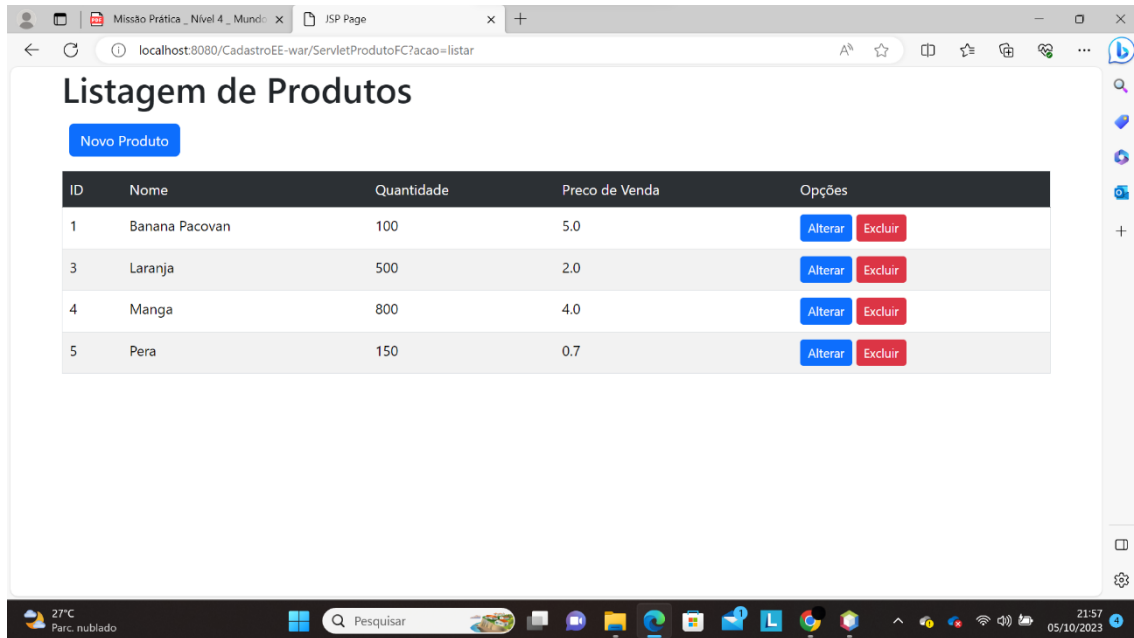
[Novo Produto](#)

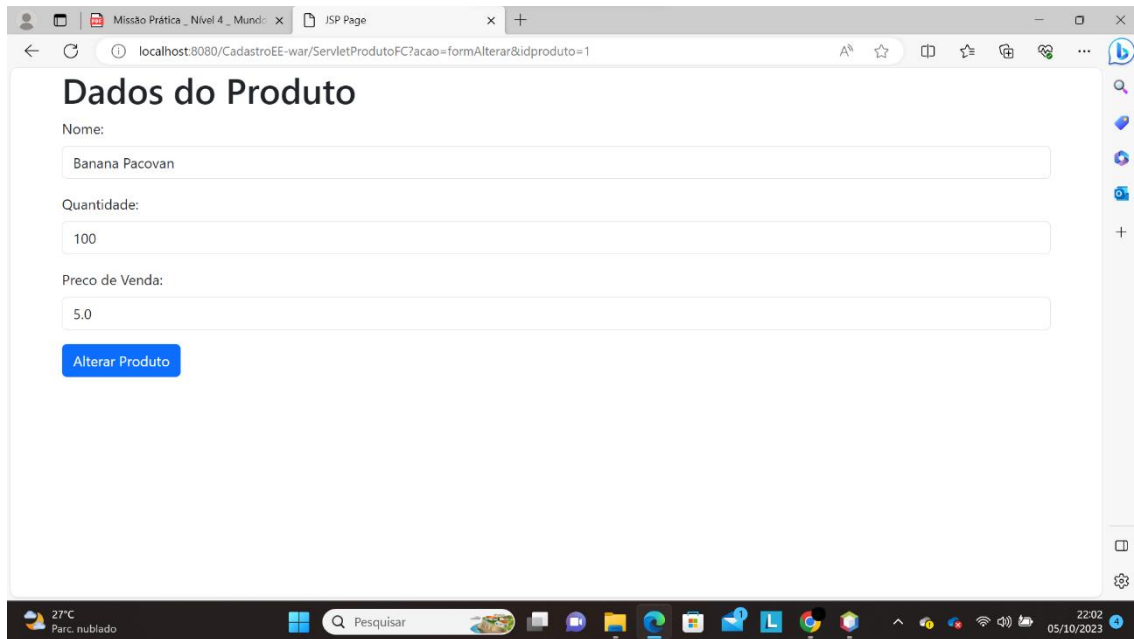
ID	Nome	Quantidade	Preço de Venda	Opções
1	Banana Pacovan	100	5.0	Alterar Excluir
3	Laranja	500	2.0	Alterar Excluir
4	Manga	800	4.0	Alterar Excluir
5	Pera	150	0.7	Alterar Excluir



► Procedimento 3: <https://github.com/Gregdev22/Missao-4-Mundo-3/tree/main/Procedimento%203>

Modificar as características de ProdutoLista.jsp e ProdutoDados.jsp por meio das bibliotecas do framework Bootstrap





Análise e Conclusão

- Como é organizado um projeto corporativo no NetBeans?

O projeto corporativo é dividido em um módulo EJB , onde ficam localizadas as classes relativas aos modelos e controles, e um módulo Web, onde ficam os arquivos JSP, HTML e Servlet.

- Qual o papel das tecnologias JPA e EJB na construção de um aplicativo para a plataforma Web no ambiente Java?

JPA --> Java Persistence API é responsável por gerenciar a interação com o banco de dados de maneira abstrata. Utilizando JPA, a persistência de dados é executada de maneira orientada a objetos. EJB --> Enterprise JavaBeans contém a lógica de negócio que atua sobre os dados de negócio.

- Como o NetBeans viabiliza a melhoria de produtividade ao lidar com as tecnologias JPA e EJB?

O desenvolvedor pode programar usando apenas conceitos de orientação a objetos, sem se preocupar com conceitos relacionais – como tabelas – ou detalhes do banco de dados.

- O que são Servlets, e como o NetBeans oferece suporte à construção desse tipo de componentes em um projeto Web?

Servlet (mini-servidor) é um objeto Java que recebe requisições do cliente (request) e produz algo (response), como uma página HTML dinamicamente gerada.

- Como funciona o padrão Front Controller, e como ele é implementado em um aplicativo Web Java, na arquitetura MVC?

É um padrão arquitetural que se comporta como um controlador tratando todas as solicitações para um site Web e então roteia para uma ação (ou comando). o Front Controller trata todas as chamadas vindas de um site web e é organizado em duas partes: através de um Manipulador Web e uma hierarquia de Comandos. O Manipulador Web é o objeto que efetivamente recebe as solicitações HTTP do tipo POST ou GET do servidor web. Ele extrai as informações necessárias da URL e das solicitações e então decide que tipo de ação iniciar e por fim delega a um objeto Comando para executar a ação.

- Quais as diferenças e semelhanças entre Servlets e JSPs?

Ambos possuem uma grande portabilidade, facilidade de programação, flexibilidade e escalabilidade. Em uma página JSP a formatação HTML se encontra separada da programação, podendo ser modificada sem afetar a aplicação de modo mais profundo. Enquanto no Servlet o resultado de uma requisição se mistura lógica de aplicação.

- Qual a diferença entre um redirecionamento simples e o uso do método forward, a partir do RequestDispatcher? Para que servem parâmetros e atributos nos objetos HttpRequest?

Redirect redireciona o cliente para uma página (sendRedirect) e RequestDispatcher encaminha uma requisição para ser atendida por outro recurso (forward). No primeiro caso (sendRedirect), o cliente receberá uma resposta http em cujo header haverá a informação de que ele deve requisitar outra página, e o browser fará esta requisição. Ou seja, o redirecionamento ocorre no lado no cliente. No segundo caso (forward), no lado do server a requisição do usuário será encaminhada para ser atendida por outro recurso (outro servlet). Este outro servlet eventualmente devolverá outra página para o usuário.

Os parâmetros são as informações da página submetidos por um formulário, por exemplo. Nesse caso, cada controle de entrada de dados é um parâmetro, e todos eles pertencem à requisição HTTP. Se você

submeter dados através do método GET, os parâmetros ficam visíveis na URL no seguinte formato: url?param1=valor1¶m2=valor2 etc. Já os atributos são objetos associados a nomes - uma espécie de tabela onde a chave é uma string - que ficam guardados no servidor, associados a um determinado escopo que pode ser de página (`PageContext`), requisição (`HttpServletRequest`), sessão (`HttpSession`) ou aplicação (`ServletContext`).

- Como o framework Bootstrap é utilizado?

O Bootstrap é um framework CSS para ser utilizado no front-end de aplicações web. Ele utiliza JavaScript e CSS para estilizar as páginas e adicionar funcionalidades através de classes específicas adicionadas as tags de um documento html.

- Por que o Bootstrap garante a independência estrutural do HTML?

O Bootstrap apenas altera o estilo da página e adiciona funcionalidades.

- Qual a relação entre o Bootstrap e a responsividade da página?

O uso do Bootstrap visa a possibilidade de acessar o site em qualquer dispositivo, pois ser responsivo é uma de suas principais características. Basicamente, ele trabalha com um sistema de grid que divide a tela em 12 colunas, que facilita a acomodação dos elementos conforme o tamanho da tela do dispositivo.