

Curso Técnico de Informática

VICTOR MANUEL

CRYSTIAN FERNANDES

MATHEUS MELLES

LUAN FILIPE

PROJETO INTEGRADOR

Rio de Janeiro – RJ

2018



Curso Técnico de Informática

VICTOR MANUEL

CRYSTIAN FERNANDES

MATHEUS MELLES

LUAN FILIPE

PROJETO INTEGRADOR

Trabalho apresentado ao Senac Bonsucesso como requisito a condução do projeto integrador, sob orientação Professor Alessandro Vieira

Rio de Janeiro – RJ

2018

**Índice**

Introdução......................................................................................................04

Cronograma...................................................................................................05

Custo do Projeto............................................................................................06

Mini Mundo....................................................................................................07

Modelo Entidade Relacionamento(M.E.R.) ..................................................08

LinguagemdeModelagemUnificada-UML......................................................09

Layout do Projeto..........................................................................................11

**Introdução**

**Entrevista inicial:**

O cliente Alessandro Vieira, presidente da empresa H&P137-infomatica, informou que deseja que seu site possua primeiramente uma área inteiramente focada na parte de venda de seus produtos com foco em mostrar principalmente a imagem do produto em questão. O cliente também requisitou que o site tivesse um cadastro para a venda do produto e que com esse cadastro o cliente possa ver todas as compras feitas em seu site.

**Justificativa:**

As vantagens e benefícios que o projeto irá proporcionar:

* O site de venda de computadores irá beneficiar as vendas de seus hardwares e periféricos
* O site irá proporcionar um aumento de visualização de sua empresa

Objetivos:

* O site fornecerá um acesso mais rápido para efetuar a compra dos seus produtos
* Para efetuar compras nesse site é obrigatório efetuar um cadastro.
* O objetivo específico será vender computadores e periféricos.

**Cronograma:**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Cronograma do Site H&P 137 – Informatica:** | | |  |  | | | | **22/05/2018** | **ATE** | **03/07/2018** |
| **Fase** | **Atividades** | **Semana/Ano** | | | | | |  |  |  |
| **1º semana** | | | **2º semana** | **3º semana** | **4º semana** | **5º semana** | **6º semana** | **7º semana** |
| **Terça – Quinta** | | | **Terça - Quinta** | **Terça - Quinta** | **Terça - Quinta** | **Terça - Quinta** | **Terça - Quinta** | **Terça** |
| Introdução | Documentação |  | | |  |  |  |  |  |  |
| Cronograma |  | | |  |  |  |  |  |  |
| Custos do Projeto |  | | |  |  |  |  |  |  |
| Minimundo |  | | |  |  |  |  |  |  |
| Modelagem | Modelo Entidade Relacionamento - MER |  | | |  |  |  |  |  |  |
| Diagrama de Caso de Uso – UML |  | | |  |  |  |  |  |  |
| Diagrama de Classes - UML |  | | |  |  |  |  |  |  |
| Desenvolvimento | Site - HTML |  | | |  |  |  |  |  |  |
| Banco de Dados |  | | |  |  |  |  |  |  |
| Java Web |  | | |  |  |  |  |  |  |

**Custo do projeto:**

**Hardware:**

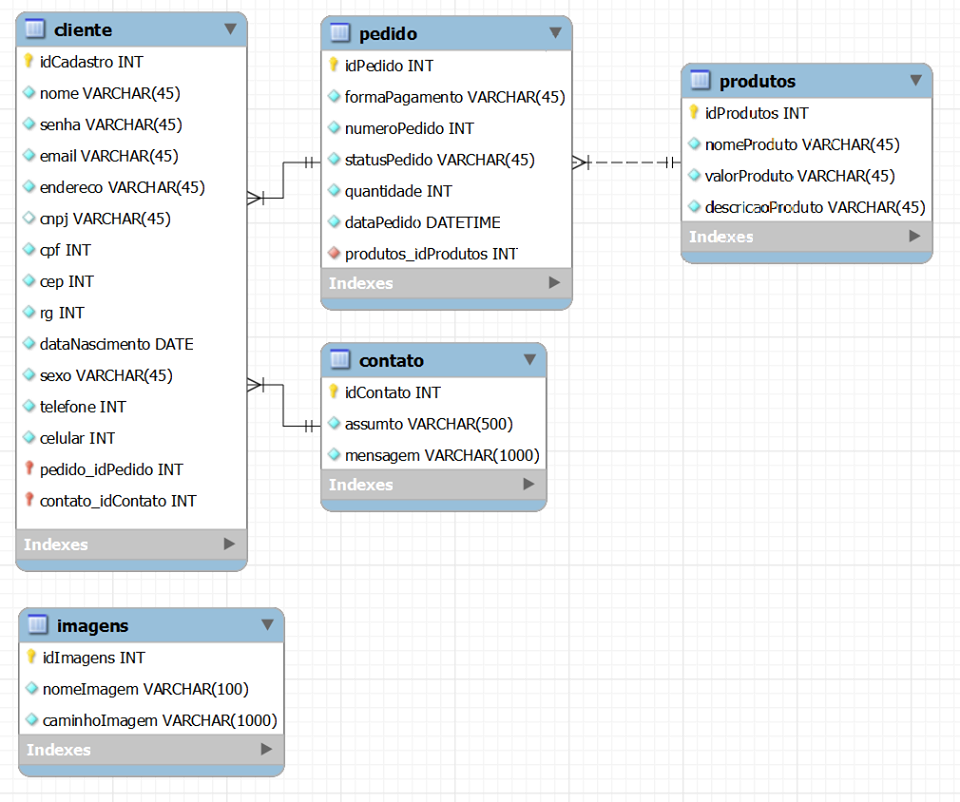
* Intel processador xeon dual core 2.3 ghz
* 500 gb disco rígido
* 4gb de memória ddr3
* 100mbps uplink
* 10 tb de transferência/mês
* 2 IPs dedicados
* cPanel gratuito

**Minimundo:**

Como pedido pelo presidente Alessandro Vieira, dono da empresa H&P137-informatica, iremos construir um site que atenderá ao pedido da criação de um cadastro para os usuários do site, armazenando todas as informações das compras em um banco de dados no servidor e o presidente poderá acessar e ver todos os registros do site.

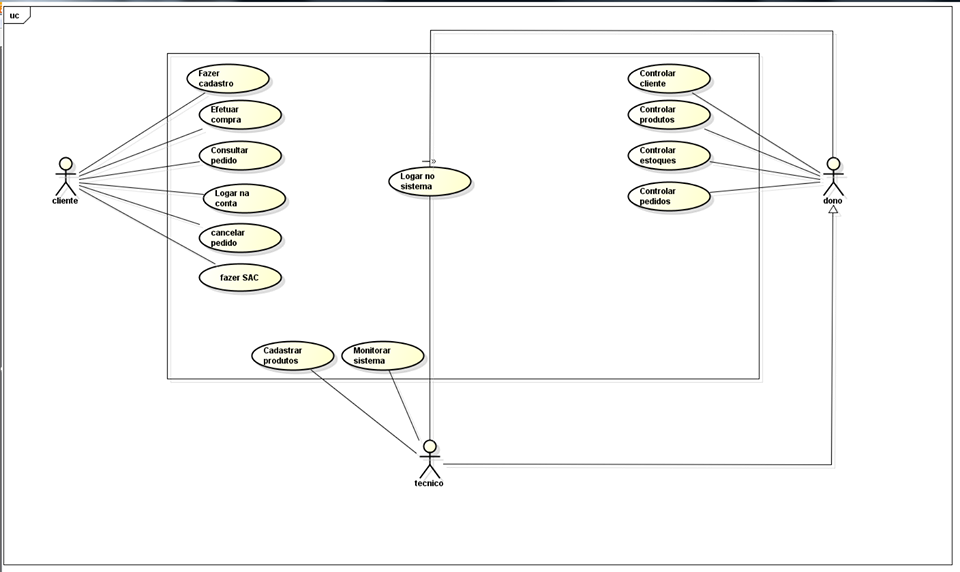
O site terá uma parte dedicada para expor os produtos que estão disponíveis para venda e assim após o cadastro do cliente e o login ele poderá efetuar a compra dos produtos. O site na página principal terá um banner com descontos de produtos e um menu de vendas de hardware e outro menu com vendas de periféricos.

Modelo Entidade Relacionamento(M.E.R.)

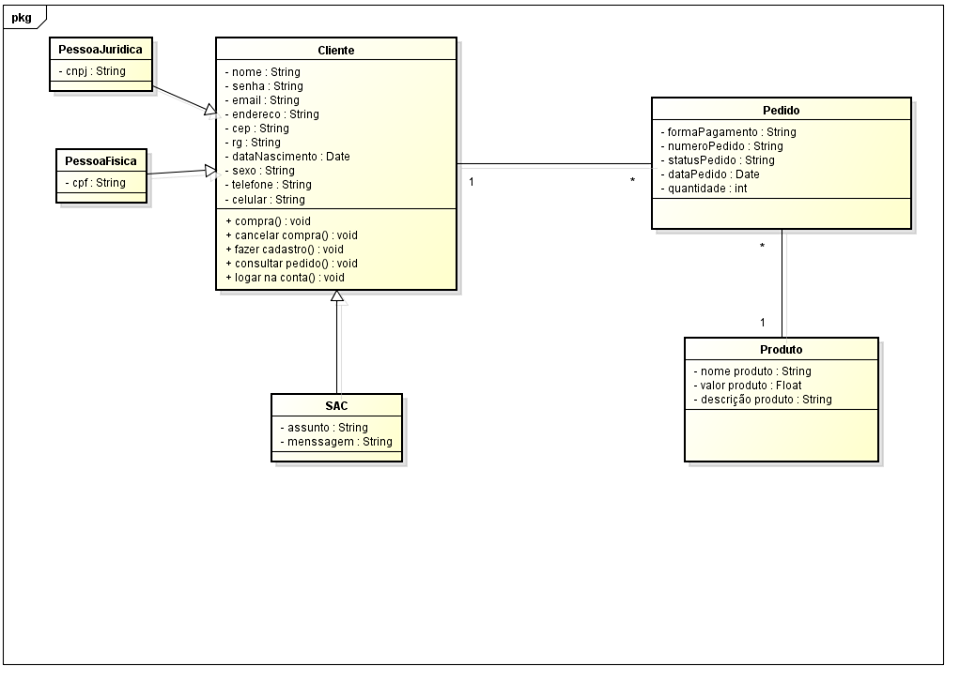


**Linguagem de Modelagem Unificada-UML**

**Diagrama de Caso de Uso**

****

**Diagrama de Classes**

****

Layout do Projeto

****

****

**sql**

-- MySQL Script generated by MySQL Workbench

-- 06/14/18 19:33:49

-- Model: New Model Version: 1.0

SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;

SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;

SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='TRADITIONAL,ALLOW\_INVALID\_DATES';

-- -----------------------------------------------------

-- Schema HePInformatica

-- -----------------------------------------------------

CREATE SCHEMA IF NOT EXISTS `HePInformatica` DEFAULT CHARACTER SET utf8 COLLATE utf8\_general\_ci ;

USE `HePInformatica` ;

-- -----------------------------------------------------

-- Table `HePInformatica`.`produtos`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `HePInformatica`.`produtos` (

`idProdutos` INT NOT NULL AUTO\_INCREMENT,

`nomeProduto` VARCHAR(45) NOT NULL,

`valorProduto` VARCHAR(45) NOT NULL,

`descricaoProduto` VARCHAR(45) NOT NULL,

PRIMARY KEY (`idProdutos`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `HePInformatica`.`pedido`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `HePInformatica`.`pedido` (

`idPedido` INT NOT NULL AUTO\_INCREMENT,

`formaPagamento` VARCHAR(45) NOT NULL,

`numeroPedido` INT NOT NULL,

`statusPedido` VARCHAR(45) NOT NULL,

`quantidade` INT NOT NULL,

`dataPedido` DATETIME NOT NULL,

`produtos\_idProdutos` INT NOT NULL,

PRIMARY KEY (`idPedido`),

INDEX `fk\_pedido\_produtos\_idx` (`produtos\_idProdutos` ASC),

CONSTRAINT `fk\_pedido\_produtos`

FOREIGN KEY (`produtos\_idProdutos`)

REFERENCES `HePInformatica`.`produtos` (`idProdutos`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `HePInformatica`.`contato`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `HePInformatica`.`contato` (

`idContato` INT NOT NULL AUTO\_INCREMENT,

`assumto` VARCHAR(500) NOT NULL,

`mensagem` VARCHAR(1000) NOT NULL,

PRIMARY KEY (`idContato`))

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `HePInformatica`.`cliente`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `HePInformatica`.`cliente` (

`idCadastro` INT NOT NULL AUTO\_INCREMENT,

`nome` VARCHAR(45) NOT NULL,

`senha` VARCHAR(45) NOT NULL,

`email` VARCHAR(45) NOT NULL,

`endereco` VARCHAR(45) NOT NULL,

`cnpj` VARCHAR(45) NULL,

`cpf` INT NOT NULL,

`cep` INT NOT NULL,

`rg` INT NOT NULL,

`dataNascimento` DATE NOT NULL,

`sexo` VARCHAR(45) NOT NULL,

`telefone` INT NOT NULL,

`celular` INT NOT NULL,

`pedido\_idPedido` INT NOT NULL,

`contato\_idContato` INT NOT NULL,

PRIMARY KEY (`idCadastro`, `pedido\_idPedido`, `contato\_idContato`),

INDEX `fk\_cliente\_pedido1\_idx` (`pedido\_idPedido` ASC),

INDEX `fk\_cliente\_contato1\_idx` (`contato\_idContato` ASC),

CONSTRAINT `fk\_cliente\_pedido1`

FOREIGN KEY (`pedido\_idPedido`)

REFERENCES `HePInformatica`.`pedido` (`idPedido`)

ON DELETE NO ACTION

ON UPDATE NO ACTION,

CONSTRAINT `fk\_cliente\_contato1`

FOREIGN KEY (`contato\_idContato`)

REFERENCES `HePInformatica`.`contato` (`idContato`)

ON DELETE NO ACTION

ON UPDATE NO ACTION)

ENGINE = InnoDB;

-- -----------------------------------------------------

-- Table `HePInformatica`.`imagens`

-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `HePInformatica`.`imagens` (

`idImagens` INT NOT NULL AUTO\_INCREMENT,

`nomeImagem` VARCHAR(100) NOT NULL,

`caminhoImagem` VARCHAR(1000) NOT NULL,

PRIMARY KEY (`idImagens`))

ENGINE = InnoDB;

SET SQL\_MODE=@OLD\_SQL\_MODE;

SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;

SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS;

**Código Java**

**conexão**

package br.com.INFORMATICA.dao;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class conexao {

private Connection con;

public Connection getConexao(){

try {

Class.forName("com.mysql.jdbc.Driver");

con = DriverManager

.getConnection("jdbc:mysql://localhost:3306/mydb",

"root",

"");

} catch (SQLException ex) {

ex.printStackTrace();

} catch (ClassNotFoundException ex) {

ex.printStackTrace();

}

return con;

}

public void fecharConexao(){

try {

con.close();

} catch (SQLException ex) {

ex.printStackTrace();

}

}

}

**clientes**

package br.com.INFORMATICA;

public class Clientes {

private int idCadastro;

private String nome ;

private String senha ;

private String email ;

private String endereco;

private String cpf ;

private String cep ;

private String rg ;

private String dataNascimento;

private String telefone;

private String celular;

private String sexo;

public String getNome() {

return nome;

}

public void setNome(String nome) {

this.nome = nome;

}

public int getIdCadastro() {

return idCadastro;

}

public void setIdCadastro(int idCadastro) {

this.idCadastro = idCadastro;

}

public String getSenha() {

return senha;

}

public void setSenha(String senha) {

this.senha = senha;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getEndereco() {

return endereco;

}

public void setEndereco(String endereco) {

this.endereco = endereco;

}

public String getCpf() {

return cpf;

}

public void setCpf(String cpf) {

this.cpf = cpf;

}

public String getCep() {

return cep;

}

public void setCep(String cep) {

this.cep = cep;

}

public String getRg() {

return rg;

}

public void setRg(String rg) {

this.rg = rg;

}

public String getDataNascimento() {

return dataNascimento;

}

public void setDataNascimento(String dataNascimento) {

this.dataNascimento = dataNascimento;

}

public String getTelefone() {

return telefone;

}

public void setTelefone(String telefone) {

this.telefone = telefone;

}

public String getSexo() {

return sexo;

}

public void setSexo(String sexo) {

this.sexo = sexo;

}

public String getCelular() {

return celular;

}

public void setCelular(String celular) {

this.celular = celular;

}

}

**produtos**

package br.com.INFORMATICA;

public class Produto {

private int id;

private String nomeProduto;

private String descricaoProduto;

private String valorProduto;

public int getId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getDescricaoProduto() {

return descricaoProduto;

}

public void setDescricaoProduto(String descricaoProduto) {

this.descricaoProduto = descricaoProduto;

}

public String getValorProduto() {

return valorProduto;

}

public void setValorProduto(String valorProduto) {

this.valorProduto = valorProduto;

}

public String getNomeProduto() {

return nomeProduto;

}

public void setNomeProduto(String nomeProduto) {

this.nomeProduto = nomeProduto;

}

}

**clienteController**

package br.com.INFORMATICA.controller;

import br.com.INFORMATICA.Clientes;

import br.com.INFORMATICA.dao.clienteDAO;

public class ClienteController {

public void incluirCliente(Clientes clientes) {

clienteDAO dao = new clienteDAO();

dao.incluir(clientes);

}

public void excluirCliente(long id) {

clienteDAO dao = new clienteDAO();

dao.excluir(id);

}

public void alterarCliente(Clientes c) {

clienteDAO dao = new clienteDAO();

dao.alterar(c);

}

}

**produtoController**

package br.com.INFORMATICA.controller;

import java.util.List;

import br.com.INFORMATICA.Produto;

import br.com.INFORMATICA.dao.produtoDAO;

public class ProdutoController {

public List<Produto> listarProduto(){

produtoDAO dao = new produtoDAO();

return dao.listar();

}

public void incluirProduto(Produto produto) {

produtoDAO dao = new produtoDAO();

dao.incluir(produto);

}

public void excluirProduto(long id) {

produtoDAO dao = new produtoDAO();

dao.excluir(id);

}

public Produto buscarProduto(long id) {

produtoDAO dao = new produtoDAO();

return dao.buscar(id);

}

public void alterarProduto(Produto p) {

produtoDAO dao = new produtoDAO();

dao.alterar(p);

}

}

**clienteDAO**

package br.com.INFORMATICA.dao;

import java.sql.PreparedStatement;

import java.sql.SQLException;

import br.com.INFORMATICA.Clientes;

public class clienteDAO extends conexao {

public void incluir(Clientes cliente) {

try{

String sql = "insert into clientes (nome,senha,email,endereco,cpf,cep,rg,dataNascimento,sexo,telefone,celular) "

+ "values (?,?,?,?,?,?,?,?,?,?,?)";

PreparedStatement ps = getConexao().prepareStatement(sql);

ps.setString(1, cliente.getNome());

ps.setString(2, cliente.getSenha());

ps.setString(3, cliente.getEmail());

ps.setString(4, cliente.getEndereco());

ps.setString(5, cliente.getCpf());

ps.setString(6, cliente.getCep());

ps.setString(7, cliente.getRg());

ps.setString(8, cliente.getDataNascimento());

ps.setString(9, cliente.getSexo());

ps.setString(10, cliente.getTelefone());

ps.setString(11, cliente.getCelular());

ps.execute();

ps.close();

}catch(SQLException ex) {

ex.printStackTrace();

} finally {

fecharConexao();

}

}

public void excluir(long id) {

try{

String sql = "delete from cliente where idCadastro = ?";

PreparedStatement ps = getConexao().prepareStatement(sql);

ps.setLong(1, id);

ps.execute();

ps.close();

}catch(SQLException e){

e.printStackTrace();

}finally{

fecharConexao();

}

}

public void alterar(Clientes c) {

try {

String sql = "update cliente set "

+ " nome = ?, "

+ " senha = ? "

+ " email = ? "

+ " endereco = ? "

+ " cpf = ? "

+ " cep = ? "

+ " rg = ? "

+ " dataNascimento = ? "

+ " sexo = ? "

+ " telefone = ? "

+ " celular = ? "

+ " where idCadastro = ?";

PreparedStatement ps = getConexao().prepareStatement(sql);

ps.setLong(1, c.getIdCadastro());

ps.setString(2, c.getNome());

ps.setString(3, c.getSenha());

ps.setString(4, c.getEmail());

ps.setString(5, c.getEndereco());

ps.setString(6, c.getCpf());

ps.setString(7, c.getCep());

ps.setString(8, c.getRg());

ps.setString(9, c.getDataNascimento());

ps.setString(10, c.getSexo());

ps.setString(11, c.getTelefone());

ps.setString(12, c.getCelular());

ps.execute();

ps.close();

} catch (SQLException e) {

e.printStackTrace();

} finally {

fecharConexao();

}

}

}

**produtoDAO**

package br.com.INFORMATICA.dao;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.List;

import br.com.INFORMATICA.Produto;

public class produtoDAO extends conexao {

public List<Produto> listar() {

List<Produto> lista = new ArrayList<Produto>();

try {

String sql = "select \* from produtos "

+ "order by nomeProduto";

PreparedStatement ps = getConexao().

prepareStatement(sql);

ResultSet rs = ps.executeQuery();

Produto p;

while (rs.next()) {

p = new Produto();

p.setId(rs.getInt("id"));

p.setNomeProduto(rs.getString("nomeProduto"));

p.setDescricaoProduto(rs.getString("descricaoProduto"));

p.setValorProduto(rs.getString("valorProduto"));

lista.add(p);

}

} catch(SQLException e) {

e.printStackTrace();

}

return lista;

}

public void incluir(Produto produto) {

try{

String sql = "insert into produtos (nomeProduto,descricaoProduto,valorProduto) "

+ "values (?,?,?)";

PreparedStatement ps = getConexao().prepareStatement(sql);

ps.setString(1, produto.getNomeProduto());

ps.setString(2, produto.getDescricaoProduto());

ps.setString(3, produto.getValorProduto());

ps.execute();

ps.close();

}catch(SQLException ex) {

ex.printStackTrace();

} finally {

fecharConexao();

}

}

public void excluir(long id) {

try{

String sql = "delete from produtos where id = ?";

PreparedStatement ps = getConexao().prepareStatement(sql);

ps.setLong(1, id);

ps.execute();

ps.close();

}catch(SQLException e){

e.printStackTrace();

}finally{

fecharConexao();

}

}

public Produto buscar(long id) {

Produto p = new Produto();

try {

String sql = "SELECT \* FROM produtos al"

+ " where al.id = ?";

PreparedStatement ps = getConexao()

.prepareStatement(sql);

ps.setLong(1, id);

ResultSet rs = ps.executeQuery();

if (rs.next()){

p.setId(rs.getInt("id"));

p.setNomeProduto(rs.getString("nomeProduto"));

p.setDescricaoProduto(rs.getString("descricaoProduto"));

p.setValorProduto(rs.getString("valorProduto"));

}

} catch(SQLException e) {

e.printStackTrace();

} finally {

fecharConexao();

}

return p;

}

public void alterar(Produto p) {

try {

String sql = "update produtos set "

+ " nomeProduto = ?, "

+ " descricaoProduto = ? "

+ " valorProduto = ? "

+ " where id = ?";

PreparedStatement ps = getConexao()

.prepareStatement(sql);

ps.setString(1, p.getNomeProduto());

ps.setString(2, p.getDescricaoProduto());

ps.setString(3, p.getValorProduto());

ps.setLong(4, p.getId());

ps.execute();

ps.close();

} catch (SQLException e) {

e.printStackTrace();

} finally {

fecharConexao();

}

}

}

CadastroCliente

package br.com.INFORMATICA.servlet;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import br.com.INFORMATICA.Clientes;

import br.com.INFORMATICA.controller.ClienteController;

@WebServlet("/cadastroCliente")

public class cadastroCliente extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

processar(request, response);

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

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

processar(request, response);

}

private void processar(HttpServletRequest request,HttpServletResponse response) throws ServletException, IOException {

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

if(acao.equalsIgnoreCase("inicioInclusao")) {

Clientes cliente = new Clientes();

request.setAttribute("produto",cliente);

request.setAttribute("acao", "incluir");

RequestDispatcher rd = request

.getRequestDispatcher("cadastro.html");

rd.forward(request, response);

}

if(acao.equalsIgnoreCase("incluir")) {

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

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

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

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

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

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

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

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

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

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

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

Clientes cliente = new Clientes();

cliente.setNome(nome);

cliente.setSenha(senha);

cliente.setEmail(email);

cliente.setEndereco(endereco);

cliente.setCpf(cpf);

cliente.setCep(cep);

cliente.setRg(rg);

cliente.setDataNascimento(dataNascimento);

cliente.setTelefone(telefone);

cliente.setCelular(celular);

cliente.setSexo(sexo);

ClienteController controller = new ClienteController();

controller.incluirCliente(cliente);

RequestDispatcher rd = request.getRequestDispatcher("cliente.html");

rd.forward(request, response);

}

if (acao.equalsIgnoreCase("excluir")) {

int id = Integer.parseInt(request.getParameter("idCadastro"));

ClienteController controller = new ClienteController();

controller.excluirCliente(id);

RequestDispatcher rd = request.getRequestDispatcher("cliente.html");

rd.forward(request, response);

}

if(acao.equalsIgnoreCase("alterar")) {

int idCadastro = Integer.parseInt(request.getParameter("idCadastro"));

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

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

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

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

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

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

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

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

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

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

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

Clientes cliente = new Clientes();

cliente.setIdCadastro(idCadastro);

cliente.setNome(nome);

cliente.setSenha(senha);

cliente.setEmail(email);

cliente.setEndereco(endereco);

cliente.setCpf(cpf);

cliente.setCep(cep);

cliente.setRg(rg);

cliente.setDataNascimento(dataNascimento);

cliente.setTelefone(telefone);

cliente.setCelular(celular);

cliente.setSexo(sexo);

ClienteController controller = new ClienteController();

controller.alterarCliente(cliente);

RequestDispatcher rd = request.getRequestDispatcher("cliente.html");

rd.forward(request, response);

}

}

}

**CadastroProduto**

package br.com.INFORMATICA.servlet;

import java.io.IOException;

import javax.servlet.RequestDispatcher;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import br.com.INFORMATICA.Produto;

import br.com.INFORMATICA.controller.ProdutoController;

@WebServlet("/cadastroProduto")

public class cadastroProduto extends HttpServlet {

private static final long serialVersionUID = 1L;

public cadastroProduto() {

super();

// TODO Auto-generated constructor stub

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

// TODO Auto-generated method stub

processar(request, response);

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)

\*/

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

processar(request, response);

}

private void processar(HttpServletRequest request,HttpServletResponse response) throws ServletException, IOException {

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

if(acao.equalsIgnoreCase("inicioInclusao")) {

Produto produto = new Produto();

request.setAttribute("produto",produto);

request.setAttribute("acao", "incluir");

RequestDispatcher rd = request

.getRequestDispatcher("CadastrarProduto.jsp");

rd.forward(request, response);

}

if(acao.equalsIgnoreCase("incluir")) {

String nomeProduto = request.getParameter("txtNomeProduto");

String descricaoProduto = request.getParameter("txtDescricaoProduto");

String valorProduto = request.getParameter("txtValorProduto");

Produto produto = new Produto();

produto.setNomeProduto(nomeProduto);

produto.setDescricaoProduto(descricaoProduto);

produto.setValorProduto(valorProduto);

ProdutoController controller = new ProdutoController();

controller.incluirProduto(produto);

RequestDispatcher rd = request.getRequestDispatcher("administrador.jsp");

rd.forward(request, response);

}

if (acao.equalsIgnoreCase("excluir")) {

int id = Integer.parseInt(request.getParameter("id"));

ProdutoController controller = new ProdutoController();

controller.excluirProduto(id);

RequestDispatcher rd = request.getRequestDispatcher("administrador.jsp");

rd.forward(request, response);

}

if(acao.equalsIgnoreCase("buscar")) {

int id = Integer.parseInt(request.getParameter("id"));

ProdutoController controller = new ProdutoController();

Produto produto = controller.buscarProduto(id);

request.setAttribute("produto",produto);

request.setAttribute("acao", "alterar");

RequestDispatcher rd = request

.getRequestDispatcher("CadastrarProduto.jsp");

rd.forward(request, response);

}

if(acao.equalsIgnoreCase("alterar")) {

int id = Integer.parseInt(request.getParameter("id"));

String nomeProduto = request.getParameter("txtNomeProduto");

String valorProduto = request.getParameter("txtValorProduto");

String descricaoProduto = request.getParameter("txtDescricaoProduto");

Produto produto = new Produto();

produto.setId(id);

produto.setNomeProduto(nomeProduto);

produto.setDescricaoProduto(descricaoProduto);

produto.setValorProduto(valorProduto);

ProdutoController controller = new ProdutoController();

controller.alterarProduto(produto);

RequestDispatcher rd = request.getRequestDispatcher("administrador.jsp");

rd.forward(request, response);

}

}

}