\*Entities

Usuário

Veiculo

Valet

Métodos contrutores, get e set

\*dão(crud)

UsuarioDAO

//Metodo de consultar USUARIO ATRAVES DO EMAIL E SENHA DELE

Consultar Usuario()

Public Usuario consultarUsuario(String email,String senha)throws Exception{

Stmt=com.prepareStatement(“select \* from Tb\_usuario where email=? And senha=?”);

Stmt.setString(1,email);

Stmt.setString(2,senha);

RS=stmt.executeQuery();

IF(RS.next()){

Return new Usuario(

RS.getInt(“id\_usuario”),

RS.getString(“nome”),

RS.getString(“email”),

RS.getString(“senha”));

Else{

Return new Usuario();

}

VALETDAO

//metodo para consultar todo o pátio através de um select enquanto a saída for nula

Mas para buscar um valet é necessario criar o objeto Veiculo para isso foi criado o objeto veiculo e valet juntos e adicionado a um arrayList de valets

ConsultarPatio()

Public ArrayList<Valet> consultarPatio(){

ArrayList<Valet>Valets=new ArrayList<>();

Stmt=con.prepareStatement(“select \* from valet where saída is null”);

RS=stmt.executeQuery();

IF(RS.next()){  
Int id=(RS.getInt(“id\_valet”);

String marca=RS.getString(“marca”);

String modelo=RS.getString(“modelo”);

String placa=RS.getString(“placa”);

Date entrada=RS.getDate(“entrada”);

Valet v=new Valet(id(new Veiculo(marca,modelo,placa)entrada);

Valets.add(v);

Return valets;

}

registrarEntrada()

//Metodo que registra a entrada do veiculo ou seja o valet é inciado, fazer insert no banco

Public void registrarEntrada(Valet valet){

Stmt=com.prepareStatement(“insert into valet(id,marca,modelo,placa,entrada) values(?,?,?,?,?)”;

Stmt.setInt(1,this.proximoId());

Stmt.setString(2,v.getVeiculo.getMarca());

Stmt.setStrign(3,v.getVeiculo.getModelo());

Stmt.setString(4,v.getVeiculo.getPlaca());

Stmt.setDate(5,new Java.sql.Date(v.getEntrada.getTime());

Stmt.executeUpdate();

ProximoID()

Private int id=0;

stmt=con.prepareStatement(“select \* from valet”);

RS=stmt.executeQuery();

While(RS.next()){

Id++;

}

Return id+1;

}

ConsultarPorID()

//método para consultar o valet inteiro a partir de um ID

Public Valet consultarPorId(int id){

Stmt=con.prepareStatement(“select \* from Valet where id=?”);

Stmt.setInt(1,id);

RS=stmt.executeQuery();

IF(RS.next()){

Id=RS.getInt(“id”);

String marca=RS.getString(“marca”);

String modelo=RS.getString(“modelo”);

String placa=Rs.getString(“placa”);

Date entrada=RS.getDate(“entrada”);

Date saída=Rs.getDate(“saída”);

Valet v=new Valet(id(new Veiculo(marca,modelo,placa)entrada);

v.setSaida(saída);

return v;

}

RegistrarSaida()

//Atraves de um id vamos atualizar determiando valet ,atualiza a tabela colocando o horário de saida para isso devemos colocar neste método o método de consultar por id que nos retorna todo as informações do valet

Public Valet registrarSaida(int id){

Stmt=con.prepareStatement(“update valet set saída=? Where id=?”);

Stmt.setInt(2,id);

Stmt.setDate(1,new Java.sql.Date(new Date().getTime()));

Valet v=new Valet();

V=valet.consultarPorId(id);

\*\*USUARIOBO

Valida o método da DAO

O método da dao é consultar usuário na qual se passa um email e senha e retorna o objeto Usuario

Public Usuario validaLogin(String email,String senha){

Return new UsuarioDAO().consultarUsuario();

}

\*\*VALETBO

//O metodo consultarPatio no valetDAO pega as informacoes do banco de valet e coloca num arrayList de valet, na valetBo devemos ver se este arrayList esta nulo ou não

ConsultarPatio()

Public ArrayList<Valet>ConsultarPatio(){

ArrayList<Valet>VALETS=new ArrayList<>();

IF(VALETS.isEmpty()){

Return null;}

Else{

Return VALETS;

}}

RegistrarVeiculo()

//Neste metodo registramos o veiculo a partir de um objeto veiculo

Public void registrarVeiculo(Veiculo v){

New ValetDAO().registrarEntrada(v,new Date());

}

RegistrarSaida()

//retornar um valet com o método do dão e setar neste objeto o preço que esta em outro método da classe ValetBO

Public Valet registrarSaida(int id){

Valet v=ValetDAO().registrarSaida(id);

v.setPreco(ValetBO.calcularPagamento();

return v;

}

Calcular Preco()

//metodo do professor

WEB CONTENT

INDEX(jsp)

<form action=”login” method=”post”>

<digite seu email><digite sua senha>

<button>

Na nossa pagina índex vamos ter um cadastro para email e senha, após isso seremos redirecionados para uma pagina servlet de login

SERVLET LOGIN

Método DoPost

String email=req.getParameter(“email”);

String senha=req.getParameter(“senha”);

Usuario u=new Usuario(email,senha);

UsuarioBO bo=new UsuarioBO();

Bo.validaLogin();

IF(u!=null){

HttpSession s=req.getSession();

S.setAttribute(“usuário”,u);

Req.setAttribute(“valets”,new ValetBO().consultarPatio());

RequestDispatcher d=req.getRequestDispatcher(“valet.jsp”);

d.forward(req,resp);

}else{

Req.setAttribute(“erro”,”Usuario não existe”);

RequestDispatcher d=req.getRequestDispatcher(“erro.jsp”);

d.forward(req,resp);

}}

ERROS.JSP

<h1>erro:

<p> ${erro}</p>

<a href=”índex.jsp”>Voltar para índex</a>

VALET.JSP

//Colocar as taglib(jstl)

<%@ taglib prefix=”c” uri=<HTTP://java.sun.com/jsp/jstl/core> %>

<c:choose>

<c:when test=”$={not empty usuário}”>

<p>Seja bem vindo, ${usuário.nome} </p>

<fieldset>

<form action=”registrar\_entrada” method=”post”>

<input dos DADOS DO VEICULO>

</form>

<c:if test=”${not empty valet}”>

<p>Veiculo: ${valet.veiculo.marca}</p>

<p>Modelo: ${valet.veiculo.modelo}</p>

<p>Placa: ${valet.veiculo.placa}</p>

<p>Entrada:${valet.entrada}</p>

<p>Saida: ${valet.saida}</p>

<p>Pagou:${valet.preco}</p>

</c:if>

<p>Estacionados</p>

<table>

<th>Marca</th>

<th>Modelo</th>

<th>Placa</th>

<th>Entrada</th>

<th>Acoes</th>

<c:forEach var=”i” items=”${valets}”>

<tr>

<td>${i.veiculo.marca}</td>

<td>${i.veiculo.modelo}</td>

<td>${i.veiculo.placa}</td>

<td>${i.veiculo.entrada}</td>

<td><a href=”registrar\_saida?id=”${valet.id}”>Efetuar saída</a>

</c:forEach>

</c:when>

<c:otherwise>

<P>you shall not pass</p>

</c:otherwise>

</c:choose>

SERVLET REGISTRAR ENTRADA E REGISTRAR SAIDA

UM METODO POST E OUTRO GET

DOPOST{

String marca=req.getParameter(“marca”);

String modelo=req.getParameter(“modelo”)

String placa=req.getParameter(“placa”);

Veiculo v=new Veiculo(marca,modelo,placa);

If(v!=null){

HttpSession s=req.getSession();

s.setAttribute(“veiculo”,v);

req.setAtribute(“valets”, new ValetBO.consultarPatio());

RequestDispatcher d=req.getRequestDispatcher(“valet.jsp”);

d.forward(req,resp);

Doget(){

String id=req.getParameter(“id”);

Valet bo=new ValetBO();

Valet v=new Valet();

Bo.registrarSaida(id);

Req.setAtribute(“valet”,v);

Req.setAtribute(“valets”, bo.consultarPatio()):

RequestDispatcher(“valet.jsp”).forward(req,resp);