

# ConnectCar

## PortoSeguro

**Integrantes:**

**Julia Vasconcelos Oliveira – RM558785 1TDSPF**

**Leonardo Cadena – RM557528 1TDSPF**

**Leonardo Barbosa – RM558230 1TDSPPO**

## **SUMARIO**

<b>1</b>	<b>..... Descrição do Projeto</b>
<b>2</b>	<b>..... Funcionalidades do Projeto</b>
<b>3</b>	<b>..... Protótipo do Projeto</b>
<b>5</b>	<b>..... Requisições do Projeto</b>

## Descritivo

O projeto "ConectCar" apresenta uma solução inovadora para o mercado de reparação automotiva, enfocando a experiência digital e a eficiência dos serviços prestados. A proposta central é a criação de um sistema integrado que permite aos clientes do Centro Automotivo Porto a realização de diagnósticos veiculares e a contratação de serviços de reparo de forma online, sem a necessidade de deslocamento até a oficina mecânica para a primeira avaliação. A iniciativa busca superar o modelo tradicional e conservador do setor, onde a interação física é um pré-requisito para qualquer diagnóstico ou serviço. O diferencial trazido pelo projeto é o uso de Inteligência Artificial para fornecer diagnósticos assertivos, ágeis e confiáveis, garantindo ao cliente uma experiência denominada "phygital" — uma junção do físico com o digital.

Este sistema não somente melhora a comodidade para o usuário, mas também otimiza o fluxo de trabalho da oficina ao diminuir o número de visitas desnecessárias. Ao mesmo tempo, oferece uma base de dados robusta para a gestão de clientes, veículos, serviços e mecânicos, como demonstrado no esquema do banco de dados fornecido. Os elementos como veículo, cliente, serviço e diagnóstico estão inter-relacionados, permitindo um acompanhamento detalhado de cada etapa do processo, desde a chegada do veículo até a finalização do serviço.

O objetivo do sistema é não apenas fornecer um serviço mais rápido e prático, mas também construir credibilidade e eficiência que estão em falta no setor, onde muitas vezes os clientes desconfiam dos diagnósticos e serviços realizados manualmente. Com um banco de dados estruturado e um sistema interativo, o Centro Automotivo Porto busca se posicionar como uma referência no mercado, valorizando a transparência e a confiança na relação com o cliente.

Por fim, os números revelam a relevância e o potencial do mercado, com a frota de veículos de passeio brasileira envelhecendo e a demanda por serviços de manutenção crescendo proporcionalmente. A Porto Seguro, dentro deste contexto, apresenta números significativos que solidificam a importância deste projeto, mostrando um incremento em prêmios no seguro automotivo e uma frota segurada expressiva, indicativos claros da confiança e dependência dos clientes para com a empresa e seus serviços.

## Funcionalidades

- Pacote: br.com.fiap.beans - Para a criação das classes principais e dos atributos do projeto.
- Pacote: br.com.fiap.DAO - Para a criação da classe ClienteDAO, dessa forma pegando as informações do cliente para integração com banco de dados, método inserir para utilizar o insert e inserir os dados no banco de dados
- Pacote: br.com.fiap.main - Para testar as classes do pacote br.com.fiap.beans, br.com.fiap.DAO e do pacote Conexao. Criação de arrayList para armazenar os dados do mecanico, criação de for in com para exibir todos mecanicos e exibição padrao apenas com SystemOutPrintln para o resto das classes
- Pacote: br.com.fiap. conexão – Para a integração com o banco de dados
- Pacote: BO – Para criação do métodos realizados no crud de Cliente, Veiculo e Mecânico
- Pacote: br.com.fiap.excecoes - Para as exceções do projeto, atualmente contando apenas com o método de falha na conexão com o banco de dados
- Pacote: br.com.fiap.model - Para a criação da classe Endereço que sera utilizada pela API externa do viaCep
- Pacote: br.com.fiap.resource - Para criação do servidor para integração da API interna com as demais materias, possui os metodos GET, POST, PUT, DELETE das seguintes classes: clienteResource, mecanicoResource, VeiculoResource.
- Pacote: br.com.fiap.service - para a integração da api viaCep

## Protótipo

```

package br.com.fiap.main;

import java.sql.SQLException;

public class TesteAtualizarCliente {

    static String texto (String j) {
        return JOptionPane.showInputDialog(j);
    }

    static int inteiro(String j) {
        return Integer.parseInt(JOptionPane.showInputDialog(j));
    }

    static double real (String j) {
        return Double.parseDouble(JOptionPane.showInputDialog(j));
    }

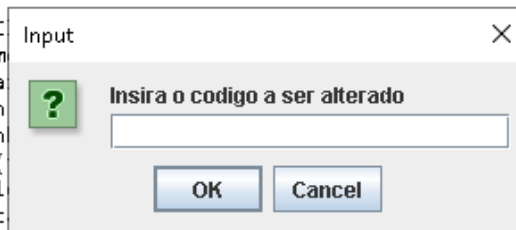
    public static void main(String[] args) throws ClassNotFoundException, SQLException {
        // Instanciar objetos
        Cliente objCliente = new Cliente();

        ClienteDAO dao = new ClienteDAO();

        objCliente.setIdC
        objCliente.setNom
        objCliente.setEma
        objCliente.setCnh
        objCliente.setSen
        objCliente.setRg(
        objCliente.setTel
        objCliente.setDat
        System.out.println(dao.atualizar(objCliente));

    }
}

```



```

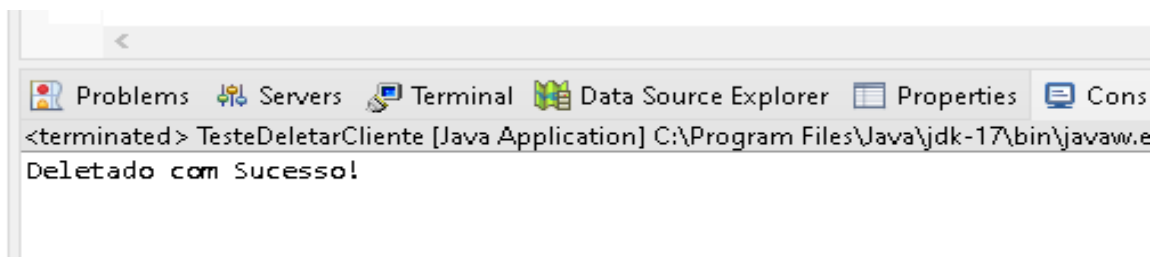
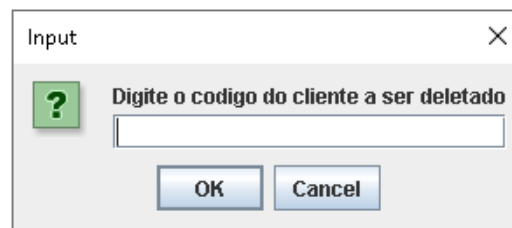
<terminated> TesteAtualizarCliente [Java App
Atualizado com Sucesso!

```

```

1 package br.com.fiap.main;
2
3 import java.sql.SQLException;
4
5 public class TesteDeletarCliente {
6
7     static int inteiro(String j) {
8         return Integer.parseInt(JOptionPane.showInputDialog(j));
9     }
10
11     public static void main(String[] args) throws ClassNotFoundException, SQLException {
12         // Instanciar objetos
13         Cliente objCliente = new Cliente();
14
15         ClienteDAO dao = new ClienteDAO();
16
17         objCliente.setIdCliente(inteiro("Digite o código do cliente a ser deletado"));
18
19         System.out.println(dao.deletar(objCliente.getIdCliente()));
20     }
21 }

```



```
1 package br.com.fazpimain;
2
3 import java.sql.Connection;
4
5
6
7
8 public class TesteConexao {
9
10     public static void main(String[] args) throws ClassNotFoundException, SQLException {
11         Connection cn = new ConexaoFactory().Conexao();
12
13         System.out.println("Conectado com banco de dados");
14
15         cn.close();
16
17     }
18 }
19 }
20
```

<

Problems Servers Terminal Data Source Explorer Properties Console X Git Staging

<terminated> TesteConexao [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe (4 de nov. de 2024 18:00:03 – 18:00:04) [pid: 8908]

Conectado com banco de dados

## Requisições

200 OK

214 ms

1650 B

Just Now ▼

Preview

Headers 7

Cookies

Tests 0 / 0

→ Mock

Console

Preview ▼

```
1 * [  
2 * {  
3   "nome": null,  
4   "email": "mecanico123",  
5   "senha": "alecarlos@gmail.com",  
6   "rg": null,  
7   "cpf": "Alexandre Carlos",  
8   "telefone": null,  
9   "dataNascimento": "5412969848",  
10  "idMecanico": 1  
11 },  
12 * {  
13   "nome": null,  
14   "email": "mecanico456",  
15   "senha": "fernandasilva@gmail.com",  
16   "rg": null,  
17   "cpf": "Fernanda Silva",  
18   "telefone": null,  
19   "dataNascimento": "12345678901",  
20   "idMecanico": 2  
21 },  
22 * {  
23   "nome": null,  
24   "email": "mecanico789",  
25   "senha": "carlosetuado@gmail.com",  
26   "rg": null,  
27   "cpf": "Carlos Eduardo",  
28   "telefone": null,  
29   "dataNascimento": "98765432100",  
30   "idMecanico": 3  
31 },  
32 * {  
33   "nome": null,  
34   "email": "mecanico321",  
35   "senha": "patricialima@gmail.com",  
36   "rg": null,  
37   "cpf": "Patrícia Lima",  
38   "telefone": null,  
39   "dataNascimento": "12312312312",  
40   "idMecanico": 4  
41 },
```



200 OK

144 ms

633 B

Just Now ▾

Preview

Headers 7

Cookies

Tests 0 / 0

→ Mock

Console

Preview ▾

```
1 - [  
2 - {  
3   "idVeiculo": 1,  
4   "placa": "1HGCMS2633A1234566",  
5   "chassi": "ABC1234"  
6 },  
7 - {  
8   "idVeiculo": 2,  
9   "placa": "1HGCMS2633A123457",  
10  "chassi": "XYZ5678"  
11 },  
12 - {  
13   "idVeiculo": 3,  
14   "placa": "1HGCMS2633A123458",  
15   "chassi": "LMN9012"  
16 },  
17 - {  
18   "idVeiculo": 4,  
19   "placa": "1HGCMS2633A123459",  
20   "chassi": "OPQ3456"  
21 },  
22 - {  
23   "idVeiculo": 5,  
24   "placa": "1HGCMS2633A123460",  
25   "chassi": "RST7890"  
26 },  
27 - {  
28   "idVeiculo": 6,  
29   "placa": "1HGCMS2633A123461",  
30   "chassi": "UWV1234"  
31 },  
32 - {  
33   "idVeiculo": 7,  
34   "placa": "1HGCMS2633A123462",  
35   "chassi": "DEF5678"  
36 },  
37 - {  
38   "idVeiculo": 8,  
39   "placa": "1HGCMS2633A123463",  
40   "chassi": "GHI9012"  
41 },  
42 - {  
43   "idVeiculo": 9,  
44   "placa": "1HGCMS2633A123464",  
45   "chassi": "JKL3456"  
46 },
```

200 OK272 ms1694 BJust Now

PreviewHeaders7CookiesTests0 / 0MockConsole

Preview

```
1 [
2   {
3     "nome": null,
4     "email": "senha1234",
5     "cnh": "Julia Vasconcelos",
6     "senha": "juliavascoli30@gmail.com",
7     "rg": null,
8     "telefone": null,
9     "dataNascimento": null,
10    "idCliente": 1,
11    "endereco": null
12  },
13  {
14    "nome": null,
15    "email": "senha5678",
16    "cnh": "Carlos Silva",
17    "senha": "carlossilva@gmail.com",
18    "rg": null,
19    "telefone": null,
20    "dataNascimento": null,
21    "idCliente": 2,
22    "endereco": null
23  },
24  {
25    "nome": null,
26    "email": "senha91011",
27    "cnh": "Mariana Oliveira",
28    "senha": "mariana.oliveira@gmail.com",
29    "rg": null,
30    "telefone": null,
31    "dataNascimento": null,
32    "idCliente": 3,
33    "endereco": null
34  },
35  {
36    "nome": null,
37    "email": "senha1213",
```

POSThttp://localhost:8080/ConnectCarSprint4/rest/veiculoSend201 Created131 ms0 B

ParamsBodyAuthHeaders4ScriptsDocsPreviewHeaders7CookiesTests0 / 0

JSONPreview

```
1 {
2   "idVeiculo": 21,
3   "placa": "21ab32i",
4   "chassi": "12dsew"
5 }
```

No body returned for response