

**UNIVERSIDADE PAULISTA – UNIP EaD**  
**Projeto Integrado Multidisciplinar**  
**Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas**

**BRUNO AURELIO S. C. RODRIGUES   RA: 0501813**

**CODIFICAÇÃO EM C# DO MECANISMO DE ACESSO A UM  
TRECHO DE BANCO DE DADOS**

Araraquara - SP  
Maio de 2021

BRUNO AURELIO S. C. RODRIGUES RA: 0501813

## **CODIFICAÇÃO EM C# DO MECANISMO DE ACESSO A UM TRECHO DE BANCO DE DADOS**

Projeto Integrado Multidisciplinar para obtenção do título de tecnólogo em Análise e Desenvolvimento de Sistemas, apresentado à Universidade Paulista – UNIP EaD.

Orientador(a): Prof. Cassiano Gunji.

Araraquara - SP

Maio de 2021

## Resumo

O Desenvolvimento de *Software* tem evoluído para além do *desktop*, sendo navegado ou melhor dizendo, acessado através de páginas *web*, de qualquer lugar do mundo.

Nesse PIM VIII, irei te mostrar como funciona um trecho de acesso ao banco de dados de uma aplicação *web*, suas interfaces para utilizar o *software* via *notebook* / computador e pelo celular, utilizando a plataforma Android.

Seguiremos em paralelo com a equipe de desenvolvimento onde foi deixado por nossa responsabilidade o acesso a um trecho do banco de dados, onde usaremos conceito de banco de dados e suas melhores práticas, a codificação, estruturação e diagramas, que podemos chamar e *back end* e auxiliaremos a equipe de *front end* na prototipação de interfaces gráficas para o ambiente Android e PC.

Palavras-Chave: Desenvolvimento, Banco de Dados, Android, ASP.NET, C#

## *Abstract*

Software development has evolved beyond the desktop, being navigated, or better said, accessed through web pages, from anywhere in the world.

In this PIM VIII, I will show you how a section of access to the database of a web application works, its interfaces for using the software via notebook / computer and cell phone, using the Android platform.

We will continue in parallel with the development team where access to a portion of the database was left to our responsibility, where we will use the database concept and its best practices, coding, structuring and diagrams, which we can call, and back end and we will assist the front end team in the prototyping of graphical interfaces for the Android and PC environment.

Keywords: Development, Database, Android, ASP.NET, C #

## LISTA DE FIGURAS

Figura 1 - <i>Stacks</i> de Desenvolvimento .....	11
Figura 2 - Logotipo Plataforma .NET .....	12
Figura 3 - Plataforma Desenvolvimento Android .....	13
Figura 4 - Tabelas do banco de dados .....	14
Figura 5 - Diagrama Entidade Relacionamento .....	15
Figura 6 - Transação / Tela Cadastro Usuário em Branco .....	17
Figura 7 - Transação / Tela Cadastro Usuário - Campos Preenchidos .....	18
Figura 8 - Transação / Tela Editar Dados .....	19
Figura 9 - Transação / Tela Listagem de Usuários .....	20
Figura 10 - Transação / Tela Exibir Detalhes do Usuário .....	21
Figura 11 - Transação / Tela Deletar Usuário .....	22
Figura 12 - Protótipo Mobile Navegação .....	24
Figura 13 - Protótipo Mobile Cadastro .....	25
Figura 14 - Protótipo Mobile Consulta .....	26

## LISTA DE CÓDIGOS FONTES

Código 1 - Solução Completa MVC .....	27
Código 2 - <i>Index</i> 1/2 .....	28
Código 3 - <i>Index</i> 2/3 .....	28
Código 4 - <i>Index</i> 3/3 .....	29
Código 5 – Criar Usuário 1/4 .....	30
Código 6 – Criar Usuário 2/4 .....	30
Código 7 - Criar Usuário 3/4 .....	31
Código 8 - Criar Usuário 4/4 .....	31
Código 9 - Editar Usuário 1/4 .....	32
Código 10 - Editar Usuário 2/4 .....	32
Código 11- Editar Usuário 3/4 .....	33
Código 12 - Editar Usuário 4/4 .....	33
Código 13 - Exibir Detalhes 1/4 .....	34
Código 14 - Exibir Detalhes 2/4 .....	34
Código 15 - Exibir Detalhes 3/4 .....	35
Código 16 - Exibir Detalhes 4/4 .....	35
Código 17 - Deletar Usuário 1/4 .....	36
Código 18 - Deletar Usuário 2/4 .....	36
Código 19 - Deletar Usuário 4/4 .....	37
Código 20 - Deletar Usuário 4/4 .....	37
Código 21 – Usuário Controller 1/5 .....	38
Código 22 - Usuário Controller 2/5 .....	38
Código 23 - Usuário Controller 3/5 .....	39
Código 24 - Usuário Controller 4/5 .....	39
Código 25 - Usuário Controller 5/5 .....	40
Código 26 - Mobile Navegação .....	41
Código 27 - Mobile Cadastro 1/6 .....	42
Código 28 - Mobile Cadastro 2/6 .....	42
Código 29 - Mobile Cadastro 3/6 .....	43
Código 30 - Mobile Cadastro 4/6 .....	44
Código 31 - Mobile Cadastro 5/6 .....	44
Código 32 - Mobile Cadastro 6/6 .....	45

Código 33 - Mobile Pesquisa 1/5 .....45

Código 34 - Mobile Pesquisa 2/5 .....46

Código 35 - Mobile Pesquisa 3/5 .....46

Código 36 - Mobile Pesquisa 4/5 .....47

Código 37 - Mobile Pesquisa 5/5 .....47

## Sumário

1. Introdução.....	9
2. O <i>Software</i> e sua trajetória .....	10
3. O Projeto .....	10
4. <i>Stacks</i> de Desenvolvimento.....	11
5. Sobre o Desenvolvimento .....	12
6. O SGBD – MySQL .....	14
7. Diagrama Entidade Relacionamento (DER) .....	15
8. Sobre o Projeto .....	16
9. Interface Gráfica do software em ASP.NET MVC.....	17
9.1. Adicionar Novo Usuário .....	17
9.2. Editando usuário .....	19
9.3. Listagem de Usuários.....	20
9.4. Detalhes do Usuário .....	21
9.5. Deletar Usuário .....	22
10. Interface para ambiente Android .....	23
10.1. Navegação .....	24
10.2. Cadastro.....	25
10.3. Campo Pesquisar .....	26
11. Solução e Códigos Fonte do <i>Software</i> .....	27
12. Código Fonte do Protótipo de Interface <i>Mobile</i> .....	41



## 1. Introdução

O Desenvolvimento de *Software* tem trazido para o mercado várias inovações para as empresas desde a indústria até o comércio local, melhorando a produtividade e qualidade em itens fabris e diminuindo cada dia a distância entre empresas e clientes. Segundo DAVENPORT e PRUSAK a única forma de se manter uma empresa entre as mais competitivas no mercado, é usar ferramentas de fácil acesso, e investir em tecnologia e inovação, dessa forma a empresa consegue se manter no mercado com um produto diferenciado, com a maximização da sua produtividade com o menor custo.

Com a ajuda de um *software* moderno e atualizado a empresa faz o gerenciamento melhor de seus recursos e aloca forças onde realmente é necessário, não desperdiçando esforços onde não atribui valor a seu produto final, obtendo a satisfação total do cliente / *stakeholders*.

## 2. O *Software* e sua trajetória

Os *softwares* estão cada dia mais presentes nas empresas, e essa mudança de paradigma começou no final do século XX início do século XXI, com a demanda cada vez feroz pela sociedade empresarial por um *software* que gerenciasses de forma eficientes suas finanças, seus estoques, e seus colaboradores.

Essa transformação pelo *software* está cada dia avançando a passos largos, onde cada pessoa possui pelo menos um aparelho celular, além de aumentar o número significativo de pessoas estarem trabalhando, estudando, utilizando aparelho como tablet, notebook e / ou um computador desktop em suas residências.

## 3. O Projeto

O projeto consiste em seu conceito, sua tipologia, etapas de afazeres e onde se é aplicado. Para isso como etapas, devemos ter o seu início, que determina o seu escopo, o meio, ou tarefas que serão desenvolvidas dentro do projeto, que poderão levar todo o seu ciclo, ou podem ter um ciclo curto, e sua entrega, que poderá ser dividida conforme cada etapa do projeto esteja finalizada, sendo denominada de entrega parcial, até juntar todas as partes, onde será feita a entrega para seu cliente.

Esse projeto tem como sua contextualização, desenvolver aspectos de um *software* que já está em desenvolvimento, onde nós iremos fazer parte desse projeto, contribuindo com as interfaces gráficas em ASP.NET que permita que o usuário faça interações com o *software* e interfaces para dispositivos móveis, com tecnologia Android / IOS, podendo ser acessados via aparelho celular ou *tablet* com esse modelo de sistemas operacional, também iremos realizar a conexão com o banco de dados utilizando um mecanismo na linguagem de programação C#.

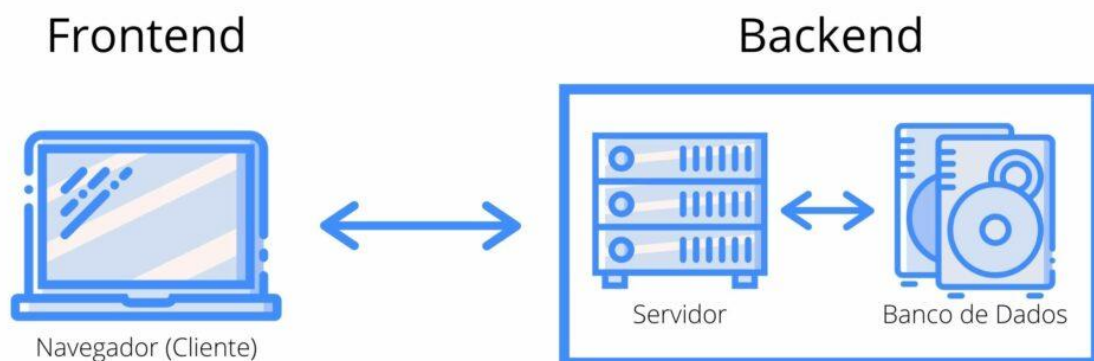
#### 4. *Stacks* de Desenvolvimento

Nos dias atuais, existem diversas formas de desenvolvimento de *software* para *web* e Android / IOS. Os mais conhecidos são os desenvolvedores de *front end* e *back end*, o desenvolvedor da forma *front end*, é responsável pela parte visível do *software*, é aquilo que o usuário vê e interage, e a forma *back end*, são os desenvolvedores que criam os dados que serão armazenados em um sistema de gerenciamento de banco de dados, que é o que o usuário não vê, que fica atrás da interface.

As linguagens e marcações que podem ser desenvolvidas para *front end* são HTML, CSS, Javascript, JQuery e fazem a utilização de diversos *frameworks* que ajudam na agilidade e praticidade do programador como Bootstrap, Materialize, entre outros.

Já a parte do *back end* fica com as linguagens e marcações de Java, ASP.NET, C#, Python, Xamarin onde o desenvolvedor cria toda lógica por trás de uma aplicação e utiliza um *software* de um gerenciamento de banco de dados para armazenar os dados gerados, inseridos pelo usuário, tais como MySQL que será usado nesse projeto e Oracle.

**Figura 1 - *Stacks* de Desenvolvimento**



Fonte: <https://marquesfernandes.com/wp-content/uploads/2020/08/Servidor-e1597608701874-1024x379.jpg>

## 5. Sobre o Desenvolvimento

Criada pela Microsoft especificamente para desenvolvimento na plataforma .NET, tem como seu principal editor de códigos o Microsoft Visual Studio, o qual usaremos para o desenvolvimento tanto para *desktop / web* como para Android/IOS.

A IDE (Ambiente de Desenvolvimento Integrado – em Português) criada pela Microsoft possibilita o desenvolvimento em várias linguagens de programação como por exemplo: C, C#, F#, C++, ASP.NET, Xamarin, Visual Basic . NET, entre outras.

Usaremos nesse projeto a linguagem ASP.NET MVC que é bem consolidada no mercado da programação e possui três pilares como sendo sua estrutura, *Web Forms*, Páginas da *Web* do ASP.NET e ASP.NET MVC. O modelo de *Web Forms*, permite a criação de *web* sites dinâmicos utilizando o método clicar e arrastar, para personalizar os componentes da página *web*, o ASP.NET permite a criação e sincronização de forma rápida com entre o *front end* e o *back end*, permitindo interações em tempo real, além de vários recursos adicionais facilitando a criação, o modelo de ASP.NET MVC é uma maneira eficiente com alto padrão de compilação de projetos *web*, utilizando metodologias ágeis, permitindo uma maior agilidade no desenvolvimento de aplicações complexas.

**Figura 2 - Logotipo Plataforma .NET**



Fonte: [https://res.cloudinary.com/practicaldev/image/fetch/s--AHIOPqi0--/c\\_imagga\\_scale,f\\_auto,fl\\_progressive,h\\_900,q\\_auto,w\\_1600/https://dev-to-uploads.s3.amazonaws.com/i/qt115zhbmh6gnyk1qluh.jpg](https://res.cloudinary.com/practicaldev/image/fetch/s--AHIOPqi0--/c_imagga_scale,f_auto,fl_progressive,h_900,q_auto,w_1600/https://dev-to-uploads.s3.amazonaws.com/i/qt115zhbmh6gnyk1qluh.jpg)

Sobre o desenvolvimento por parte de tecnologia *mobile*, Android, com o Android Studio é capaz de criar aplicativos utilizando as linguagens de programação Java e Kotlin, onde poderá realizar as compilações para plataforma Android para diversos dispositivos, como celular, televisões e tablets.

**Figura 3 - Plataforma Desenvolvimento Android**



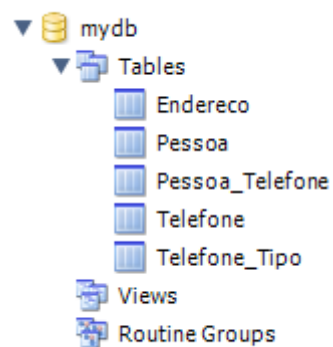
Fonte: <https://developer.android.com/studio>

## 6. O SGBD – MySQL

Para esse projeto iremos utilizar a ferramenta MySQL Workbench. Essa ferramenta utiliza linguagem SQL para realização de funções de inserção, alteração, gerenciamento e remoção, e podemos fazer a modelagem do banco de dados na própria aplicação.

Esse modelo de plataforma, utiliza banco de dados relacional e sua plataforma é Open Source, ou seja, possui seu código aberto para que diversos desenvolvedores possam contribuir aprimorando a ferramenta e seus métodos de gerenciamento aos dados.

**Figura 4 - Tabelas do banco de dados**

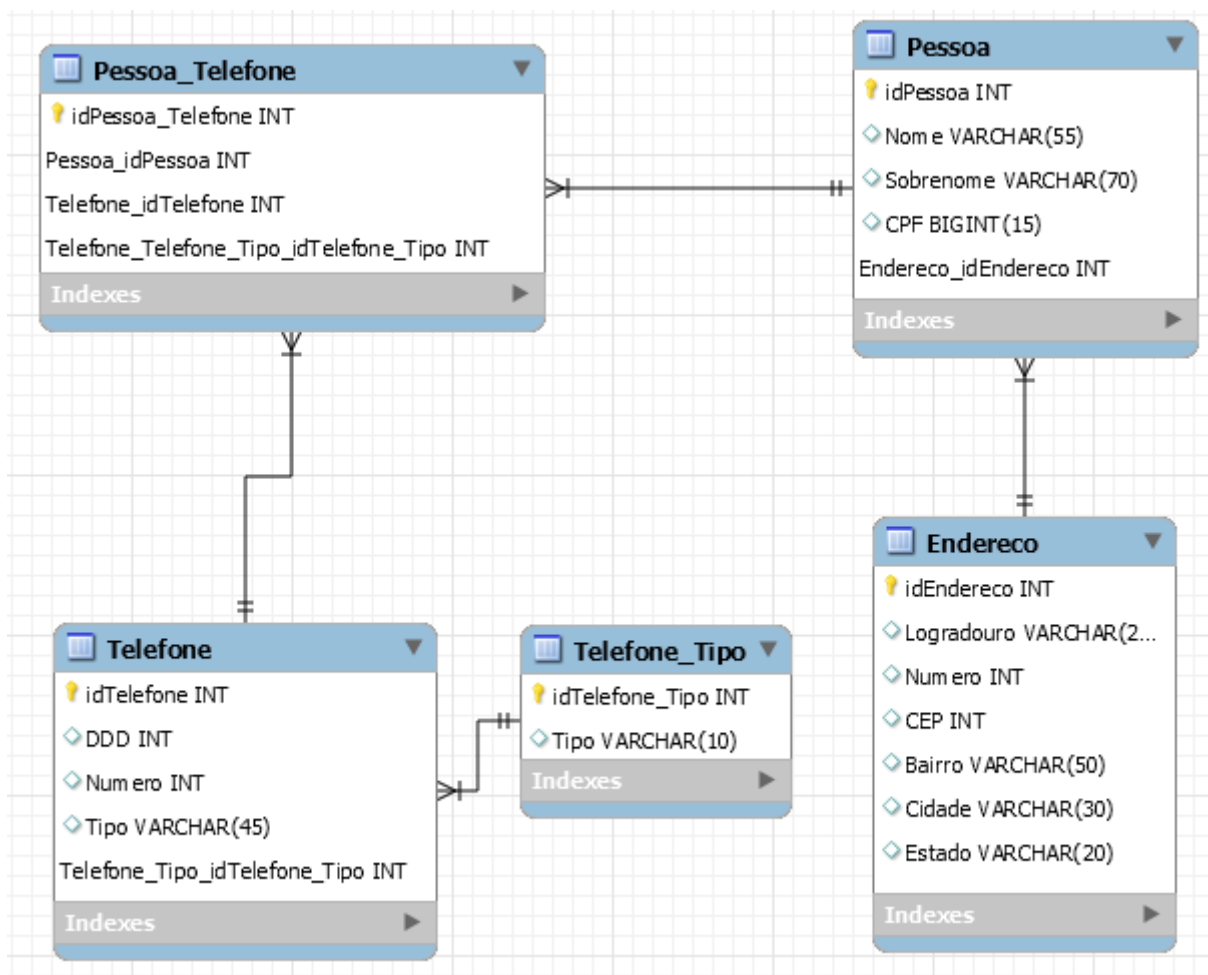


Fonte: Autoria Própria

## 7. Diagrama Entidade Relacionamento (DER)

Foi passado para a equipe responsável pelo trecho do software o seguinte modelo de entidade relacionamento a ser implantado, esse modelo foi desenvolvido utilizando o software MySQL.

**Figura 5 - Diagrama Entidade Relacionamento**



Fonte: Autoria Própria

## 8. Sobre o Projeto

O objetivo do projeto é desenvolver um mecanismo de acesso utilizando linguagem de programação em C#, utilizando o acesso ao banco de dados MySQL. A interface gráfica do *software* será em ASP.NET MVC para ambiente utilizando um computador e Android / IOS quando utilizado um aparelho celular.

Abaixo estão os objetivos desse projeto:

Desenvolver um protótipo de interface em linguagem C#, ASP.NET MVC que permita a interação do usuário, realizando inserção, alteração e remoção de dados, que serão atualizados no banco de dados.

Criação de um protótipo de interface para ambiente Android / IOS que o usuário consiga interagir da mesma forma que no ambiente utilizando um computador.

Desenvolver um mecanismo de acesso ao MySQL, que é o banco de dados utilizando nesse projeto, que será responsável por efetuar o método CRUD (*Create, Read, Update, Delete*) sigla em inglês, para Criar, Ler, Atualizar, Deletar que são operações básicas em base de dados relacional.



## 9. Interface Gráfica do software em ASP.NET MVC

Em seguida serão mostradas as principais transações / tela do sistema.

### 9.1. Adicionar Novo Usuário

Nessa transação, o formulário está em branco, aguardando a digitação dos dados.

**Figura 6 - Transação / Tela Cadastro Usuário em Branco**

#### Adicionar Usuário

<b>Nome do Usuário</b> <input type="text" value="Nome do Usuário"/>	<b>Logradouro</b> <input type="text" value="Logradouro"/>	<b>Cidade</b> <input type="text" value="Cidade"/>
<b>CPF</b> <input type="text" value="CPF"/>	<b>No.</b> <input type="text" value="Número da Residência"/>	<b>UF</b> <input type="text" value="[Selecione o estado]"/>
<b>Tipo</b> <input type="text" value="Tipo"/>	<b>Bairro</b> <input type="text" value="Bairro"/>	<b>Cadastrar Usuário</b>
<b>DDD</b> <input type="text" value="DDD"/>	<b>CEP</b> <input type="text" value="CEP"/>	
<b>Telefone</b> <input type="text" value="Telefone"/>		

Após preencher os dados com o novo usuário, clicar no botão Adicionar.

**Figura 7 - Transação / Tela Cadastro Usuário - Campos Preenchidos**

### Adicionar Usuário

<b>Nome do Usuário</b> <input type="text" value="Bruno"/>	<b>Logradouro</b> <input type="text" value="Rua"/>	<b>Cidade</b> <input type="text" value="Araraquara"/>
<b>CPF</b> <input type="text" value="00000000000"/>	<b>No.</b> <input type="text" value="111"/>	<b>UF</b> <input type="text" value="São Paulo"/>
<b>Tipo</b> <input type="text" value="Celular"/>	<b>Bairro</b> <input type="text" value="Jardim"/>	<input type="button" value="Cadastrar Usuário"/>
<b>DDD</b> <input type="text" value="16"/>	<b>CEP</b> <input type="text" value="00000000"/>	
<b>Telefone</b> <input type="text" value="000000000"/>		

## 9.2. Editando usuário

Nessa transação é possível editar um usuário que já foi cadastrado no banco de dados.

**Figura 8 - Transação / Tela Editar Dados**

[Projeto PIM VIII](#) [Lista Usuários](#) [Adicionar Novo Usuário](#) [Detalhes do Usuário](#)

## Editar

Edite os dados abaixo:

---

Nome	<input type="text" value="Bruno"/>
CPF	<input type="text" value="0000000000"/>
Tipo	<input type="text" value="Fixo"/>
DDD	<input type="text" value="16"/>
Telefone	<input type="text" value="00000000"/>
Logradouro	<input type="text" value="Rua"/>
Numero	<input type="text" value="0.00"/>
Bairro	<input type="text" value="Jardim"/>
CEP	<input type="text" value="00000000"/>
Cidade	<input type="text" value="Araraquara"/>
UF	<input type="text" value="SP"/>

Salvar

Fonte: Autoria Própria

### 9.3. Listagem de Usuários

Nessa página, podemos visualizar todos os usuários que já foram cadastrados no banco de dados do *software*.

**Figura 9 - Transação / Tela Listagem de Usuários**

Projeto PIM VIII	Lista Usuários	Adicionar Novo Usuário	Detalhes do Usuário
------------------	----------------	------------------------	---------------------

Lista de Usuários											
<a href="#">Criar Novo Usuário</a>											
Nome	CPF	Tipo	DDD	Telefone	Logradouro	Numero	Bairro	CEP	Cidade	UF	
Bruno	0000000000	Fixo	16.00	0.00	Rua	0.00	Jardim	00000000	Araraquara	SP	<a href="#">Editar</a>   <a href="#">Detalhes</a>   <a href="#">Deletar</a>
Aurelio	1111111111	Celular	16.00	0.00	Avenida	111.00	Jardim	11111111	Araraquara	SP	<a href="#">Editar</a>   <a href="#">Detalhes</a>   <a href="#">Deletar</a>
Soares	2222222222	Fixo	16.00	2222.00	Avenida	22.00	Vila	22222222	Araraquara	SP	<a href="#">Editar</a>   <a href="#">Detalhes</a>   <a href="#">Deletar</a>
Campos	3333333333	Celular	16.00	33333333.00	Rua	3333.00	Jardim	33333333	Araraquara	SP	<a href="#">Editar</a>   <a href="#">Detalhes</a>   <a href="#">Deletar</a>
Campos	4444444444	Fixo	16.00	44444444.00	Avenida	4.00	Vila	44444444	Araraquara	SP	<a href="#">Editar</a>   <a href="#">Detalhes</a>   <a href="#">Deletar</a>

© 2021 - Projeto PIM VIII - Desenvolvido por Bruno Aurelio

Fonte: Autoria Própria

## 9.4. Detalhes do Usuário

Nessa transação é possível visualizar os detalhes de cada usuário.

**Figura 10 - Transação / Tela Exibir Detalhes do Usuário**

[Projeto PIM VIII](#) [Lista Usuários](#) [Adicionar Novo Usuário](#) [Detalhes do Usuário](#)

## Detalhes

Detalhes do Usuário

---

Nome	Aurelio
CPF	11111111111
Tipo	Celular
DDD	16.00
Telefone	0.00
Logradouro	Avenida
Numero	111.00
Bairro	Jardim
CEP	11111111
Cidade	Araraquara
UF	SP

[Editar](#) | [Voltar para a Lista](#)

---

© 2021 - Projeto PIM VIII - Desenvolvido por Bruno Aurelio

Fonte: Autoria Própria

## 9.5. Deletar Usuário

Nessa transação é possível deletar / excluir o registro de um usuário selecionado do banco de dados.

**Figura 11 - Transação / Tela Deletar Usuário**

[Projeto PIM VIII](#) [Lista Usuários](#) [Adicionar Novo Usuário](#) [Detalhes do Usuário](#)

## Deletar

Tem certeza que deseja deletar esse usuário?

Detalhes do Usuário

---

Nome	Aurelio
CPF	11111111111
Tipo	Celular
DDD	16.00
Telefone	0.00
Logradouro	Avenida
Numero	111.00
Bairro	Jardim
CEP	11111111
Cidade	Araraquara
UF	SP

| [Voltar para a Lista](#)

---

© 2021 - Projeto PIM VIII - Desenvolvido por Bruno Aurelio

Fonte: Autoria Própria

## **10. Interface para ambiente Android**

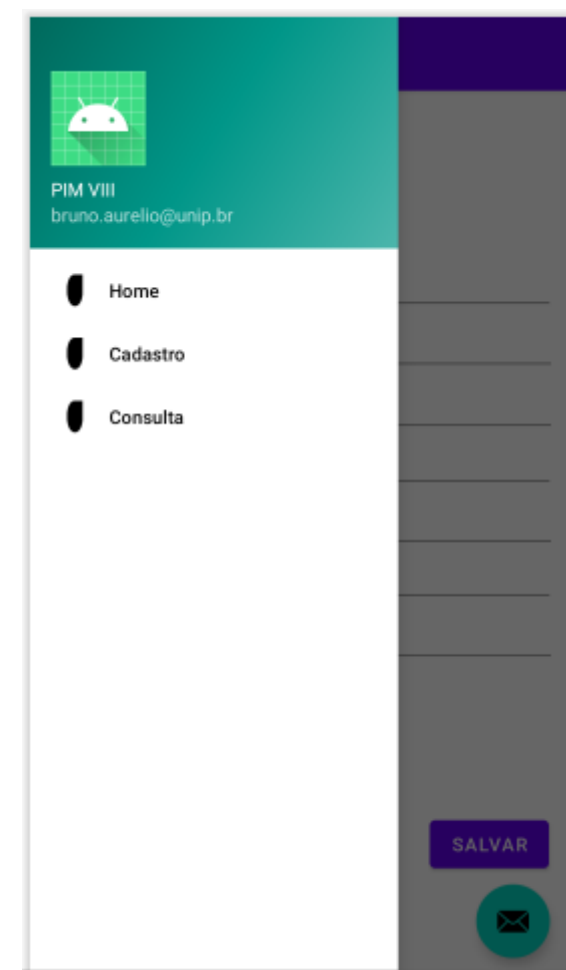
O protótipo para interface mobile foi desenvolvido pela ferramenta Android Studio em linguagem Kotlin. As interfaces não possuem ligação com software desenvolvido em ASP.NET.

Nos protótipos abaixo, pode mostrarei como fica a interface de Navegação, Cadastro e Pesquisa.

### 10.1. Navegação

A navegação é feita pela lateral esquerda do aplicativo, onde o usuário visualiza todas as funções do aplicativo.

**Figura 12 - Protótipo Mobile Navegação**



Fonte: Autoria Própria



## 10.2. Cadastro

Nessa seção o usuário realiza o cadastro preenchendo as informações que são pedidas no aplicativo e para salvar as informações, clica em Salvar.

**Figura 13 - Protótipo Mobile Cadastro**

The image shows a mobile app prototype for a registration screen titled "Novo Cadastro". The screen has a white background with a purple header bar. Below the title, there are several input fields for user information, arranged in a form-like structure. The fields are: "Nome Completo" (Full Name), "Tipo" (Type), "DDD" (Area Code), "Telefone" (Phone Number), "CPF" (Tax ID), "Logradouro" (Address), "Bairro" (Neighborhood), "No." (Number), "Cidade" (City), "CEP" (Postal Code), and "UF" (State). At the bottom right of the screen, there is a purple button labeled "SALVAR" (Save) and a circular icon with a white envelope symbol on a teal background.

Fonte: Autoria Própria

### 10.3. Campo Pesquisar

Na seção pesquisar, o usuário pode efetuar a pesquisa de duas formas, pelo nome ou cpf do usuário já cadastrado.

No exemplo abaixo mostra a pesquisa pelo nome Bruno e mostrando logo abaixo todos os usuários que contém o nome Bruno cadastrado no *software*, como o Bruno Aurelio, Bruno Soares, Bruno Campos.

**Figura 14 - Protótipo Mobile Consulta**

**Consulta**

Bruno

CPF

PESQUISAR

Q

Bruno Aurelio

Bruno Aurelio

Bruno Campos

Bruno Soares

Bruno Rodrigues

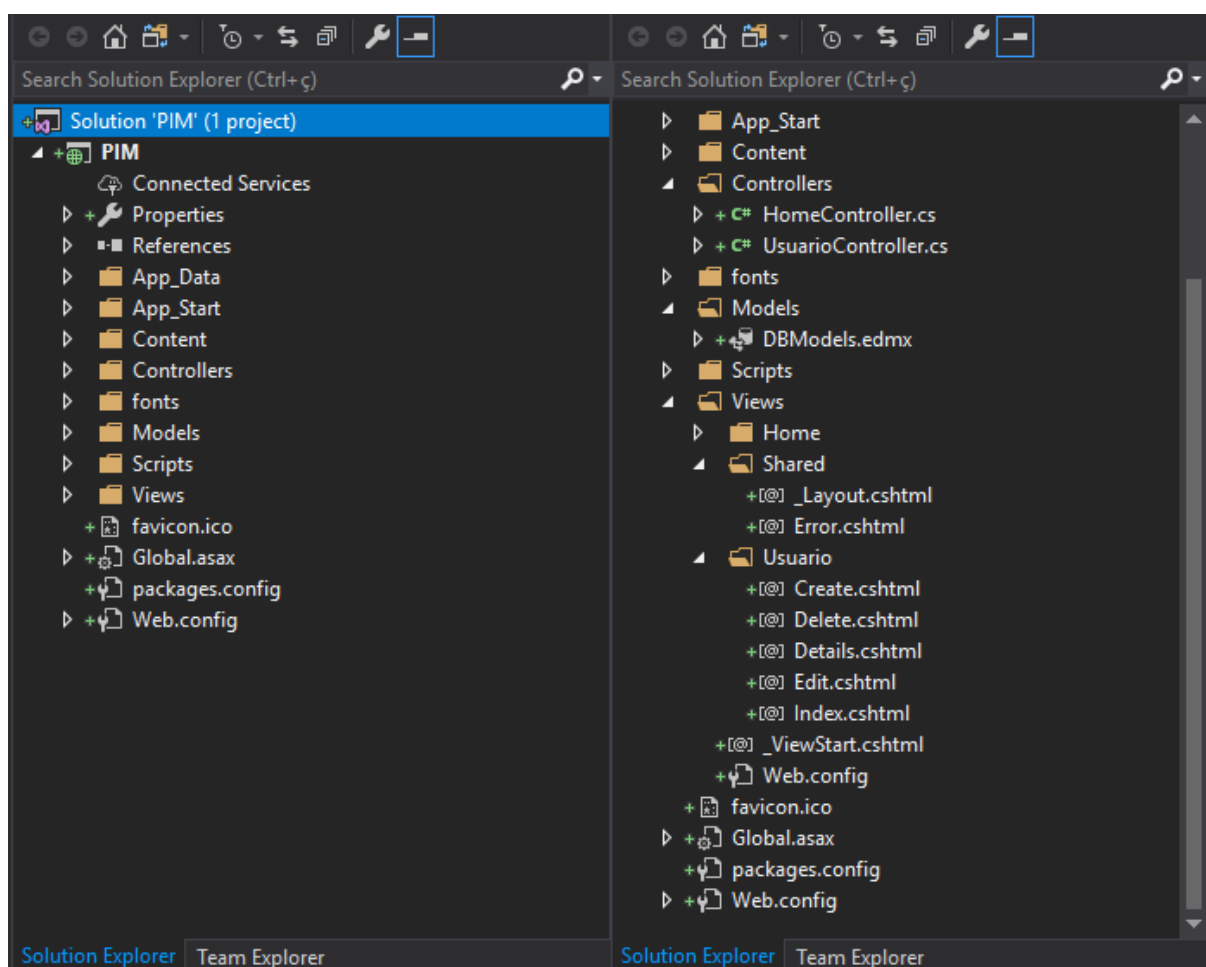
●

Fonte: Autoria Própria

## 11. Solução e Códigos Fonte do *Software*

Abaixo segue a solução criada pelo Microsoft Visual Studio no modelo MVC em linguagem de programação C# e suas subpastas com os arquivos MVC e Banco de Dados criados.

### Código 1 - Solução Completa MVC



Fonte: Autoria Própria

## Código 2 - Index 1/2

```

Index.cshtml  [X]
1  @model IEnumerable<PIM.Models.Table>
2
3  @{
4      ViewBag.Title = "Index";
5  }
6
7  <h2>Lista de Usuários</h2>
8
9  <link href="https://cdn.datatables.net/1.10.24/css/dataTables.bootstrap4.min.css" rel="stylesheet" />
10
11
12  <p>
13      @Html.ActionLink("Criar Novo Usuário", "Create")
14  </p>
15  <table id="example" class="table table-striped table-bordered" style="width:100%">
16  <tr>
17      <th>
18          @Html.DisplayNameFor(model => model.Nome)
19      </th>
20      <th>
21          @Html.DisplayNameFor(model => model.CPF)
22      </th>
23      <th>
24          @Html.DisplayNameFor(model => model.Tipo)
25      </th>
26      <th>
27          @Html.DisplayNameFor(model => model.DDD)
28      </th>
29      <th>
30          @Html.DisplayNameFor(model => model.Telefone)
31      </th>
32  </tr>
33  </table>

```

Fonte: Autoria Própria

## Código 3 - Index 2/3

```

Index.cshtml  [X]
34      @Html.DisplayNameFor(model => model.Logradouro)
35  </th>
36  <th>
37      @Html.DisplayNameFor(model => model.Numero)
38  </th>
39  <th>
40      @Html.DisplayNameFor(model => model.Bairro)
41  </th>
42  <th>
43      @Html.DisplayNameFor(model => model.CEP)
44  </th>
45  <th>
46      @Html.DisplayNameFor(model => model.Cidade)
47  </th>
48  <th>
49      @Html.DisplayNameFor(model => model.UF)
50  </th>
51  </th></th>
52  </tr>
53
54  @foreach (var item in Model)
55  {
56      <tr>
57          <td>
58              @Html.DisplayFor(modelItem => item.Nome)
59          </td>
60          <td>
61              @Html.DisplayFor(modelItem => item.CPF)
62          </td>
63          <td>
64              @Html.DisplayFor(modelItem => item.Tipo)
65          </td>
66          <td>

```

Fonte: Autoria Própria

## Código 4 - Index 3/3

```
Index.cshtml  X
67      @Html.DisplayFor(modelItem => item.DDD)
68  </td>
69  <td>
70      @Html.DisplayFor(modelItem => item.Telefone)
71  </td>
72  <td>
73      @Html.DisplayFor(modelItem => item.Logradouro)
74  </td>
75  <td>
76      @Html.DisplayFor(modelItem => item.Numero)
77  </td>
78  <td>
79      @Html.DisplayFor(modelItem => item.Bairro)
80  </td>
81  <td>
82      @Html.DisplayFor(modelItem => item.CEP)
83  </td>
84  <td>
85      @Html.DisplayFor(modelItem => item.Cidade)
86  </td>
87  <td>
88      @Html.DisplayFor(modelItem => item.UF)
89  </td>
90  <td>
91      @Html.ActionLink("Editar", "Edit", new { id = item.Id }) |
92      @Html.ActionLink("Detalhes", "Details", new { id = item.Id }) |
93      @Html.ActionLink("Deletar", "Delete", new { id = item.Id })
94  </td>
95  </tr>
96  }
97  </table>
98
99
```

Fonte: Autoria Própria

### Código 5 – Criar Usuário 1/4

```

Create.cshtml  X
1  @model PIM.Models.Table
2
3  @{
4      ViewBag.Title = "Create";
5  }
6
7  <h2>Cadastro Usuário</h2>
8
9
10 @using (Html.BeginForm())
11 {
12     @Html.AntiForgeryToken()
13
14     <div class="form-horizontal">
15         <h4>Preencha os dados abaixo:</h4>
16         <hr />
17         @Html.ValidationSummary(true, "", new { @class = "text-danger" })
18         <div class="form-group">
19             @Html.LabelFor(model => model.Nome, htmlAttributes: new { @class = "control-label col-md-2" })
20             <div class="col-md-10">
21                 @Html.EditorFor(model => model.Nome, new { htmlAttributes = new { @class = "form-control" } })
22                 @Html.ValidationMessageFor(model => model.Nome, "", new { @class = "text-danger" })
23             </div>
24         </div>
25
26         <div class="form-group">
27             @Html.LabelFor(model => model.CPF, htmlAttributes: new { @class = "control-label col-md-2" })
28             <div class="col-md-10">
29                 @Html.EditorFor(model => model.CPF, new { htmlAttributes = new { @class = "form-control" } })
30                 @Html.ValidationMessageFor(model => model.CPF, "", new { @class = "text-danger" })
31             </div>
32         </div>
33     </div>

```

Fonte: Autoria Própria

### Código 6 – Criar Usuário 2/4

```

Create.cshtml  X
34 <div class="form-group">
35     @Html.LabelFor(model => model.Tipo, htmlAttributes: new { @class = "control-label col-md-2" })
36     <div class="col-md-10">
37         @Html.EditorFor(model => model.Tipo, new { htmlAttributes = new { @class = "form-control" } })
38         @Html.ValidationMessageFor(model => model.Tipo, "", new { @class = "text-danger" })
39     </div>
40 </div>
41
42 <div class="form-group">
43     @Html.LabelFor(model => model.DDD, htmlAttributes: new { @class = "control-label col-md-2" })
44     <div class="col-md-10">
45         @Html.EditorFor(model => model.DDD, new { htmlAttributes = new { @class = "form-control" } })
46         @Html.ValidationMessageFor(model => model.DDD, "", new { @class = "text-danger" })
47     </div>
48 </div>
49
50 <div class="form-group">
51     @Html.LabelFor(model => model.Telefone, htmlAttributes: new { @class = "control-label col-md-2" })
52     <div class="col-md-10">
53         @Html.EditorFor(model => model.Telefone, new { htmlAttributes = new { @class = "form-control" } })
54         @Html.ValidationMessageFor(model => model.Telefone, "", new { @class = "text-danger" })
55     </div>
56 </div>
57
58 <div class="form-group">
59     @Html.LabelFor(model => model.Logradouro, htmlAttributes: new { @class = "control-label col-md-2" })
60     <div class="col-md-10">
61         @Html.EditorFor(model => model.Logradouro, new { htmlAttributes = new { @class = "form-control" } })
62         @Html.ValidationMessageFor(model => model.Logradouro, "", new { @class = "text-danger" })
63     </div>
64 </div>
65
66 <div class="form-group">

```

Fonte: Autoria Própria

### Código 7 - Criar Usuário 3/4

```

67      @Html.LabelFor(model => model.Numero, htmlAttributes: new { @class = "control-label col-md-2" })
68      <div class="col-md-10">
69          @Html.EditorFor(model => model.Numero, new { htmlAttributes = new { @class = "form-control" } })
70          @Html.ValidationMessageFor(model => model.Numero, "", new { @class = "text-danger" })
71      </div>
72  </div>
73
74  <div class="form-group">
75      @Html.LabelFor(model => model.Bairro, htmlAttributes: new { @class = "control-label col-md-2" })
76      <div class="col-md-10">
77          @Html.EditorFor(model => model.Bairro, new { htmlAttributes = new { @class = "form-control" } })
78          @Html.ValidationMessageFor(model => model.Bairro, "", new { @class = "text-danger" })
79      </div>
80  </div>
81
82  <div class="form-group">
83      @Html.LabelFor(model => model.CEP, htmlAttributes: new { @class = "control-label col-md-2" })
84      <div class="col-md-10">
85          @Html.EditorFor(model => model.CEP, new { htmlAttributes = new { @class = "form-control" } })
86          @Html.ValidationMessageFor(model => model.CEP, "", new { @class = "text-danger" })
87      </div>
88  </div>
89
90  <div class="form-group">
91      @Html.LabelFor(model => model.Cidade, htmlAttributes: new { @class = "control-label col-md-2" })
92      <div class="col-md-10">
93          @Html.EditorFor(model => model.Cidade, new { htmlAttributes = new { @class = "form-control" } })
94          @Html.ValidationMessageFor(model => model.Cidade, "", new { @class = "text-danger" })
95      </div>
96  </div>
97
98  <div class="form-group">
99      @Html.LabelFor(model => model.UF, htmlAttributes: new { @class = "control-label col-md-2" })

```

Fonte: Autoria Própria

### Código 8 - Criar Usuário 4/4

```

97
98  <div class="form-group">
99      @Html.LabelFor(model => model.UF, htmlAttributes: new { @class = "control-label col-md-2" })
100     <div class="col-md-10">
101         @Html.EditorFor(model => model.UF, new { htmlAttributes = new { @class = "form-control" } })
102         @Html.ValidationMessageFor(model => model.UF, "", new { @class = "text-danger" })
103     </div>
104 </div>
105
106 <div class="form-group">
107     <div class="col-md-offset-2 col-md-10">
108         <input type="submit" value="Salvar Dados" class="btn btn-default" />
109     </div>
110 </div>
111 </div>
112
113
114 <div>
115     @Html.ActionLink("Back to List", "Index")
116 </div>
117
118 @section Scripts {
119     @Scripts.Render("~/bundles/jqueryval")
120 }
121

```

Fonte: Autoria Própria

### Código 9 - Editar Usuário 1/4

```

1  @model PIM.Models.Table
2
3  @{
4      ViewBag.Title = "Edit";
5  }
6
7  <h2>Editar</h2>
8
9
10 @using (Html.BeginForm())
11 {
12     @Html.AntiForgeryToken()
13
14     <div class="form-horizontal">
15         <h4>Edite os dados abaixo:</h4>
16         <hr />
17         @Html.ValidationSummary(true, "", new { @class = "text-danger" })
18         @Html.HiddenFor(model => model.Id)
19
20         <div class="form-group">
21             @Html.LabelFor(model => model.Nome, htmlAttributes: new { @class = "control-label col-md-2" })
22             <div class="col-md-10">
23                 @Html.EditorFor(model => model.Nome, new { htmlAttributes = new { @class = "form-control" } })
24                 @Html.ValidationMessageFor(model => model.Nome, "", new { @class = "text-danger" })
25             </div>
26         </div>
27
28         <div class="form-group">
29             @Html.LabelFor(model => model.CPF, htmlAttributes: new { @class = "control-label col-md-2" })
30             <div class="col-md-10">
31                 @Html.EditorFor(model => model.CPF, new { htmlAttributes = new { @class = "form-control" } })
32                 @Html.ValidationMessageFor(model => model.CPF, "", new { @class = "text-danger" })
33             </div>

```

Fonte: Autoria Própria

### Código 10 - Editar Usuário 2/4

```

34 </div>
35
36 <div class="form-group">
37     @Html.LabelFor(model => model.Tipo, htmlAttributes: new { @class = "control-label col-md-2" })
38     <div class="col-md-10">
39         @Html.EditorFor(model => model.Tipo, new { htmlAttributes = new { @class = "form-control" } })
40         @Html.ValidationMessageFor(model => model.Tipo, "", new { @class = "text-danger" })
41     </div>
42 </div>
43
44 <div class="form-group">
45     @Html.LabelFor(model => model.DDD, htmlAttributes: new { @class = "control-label col-md-2" })
46     <div class="col-md-10">
47         @Html.EditorFor(model => model.DDD, new { htmlAttributes = new { @class = "form-control" } })
48         @Html.ValidationMessageFor(model => model.DDD, "", new { @class = "text-danger" })
49     </div>
50 </div>
51
52 <div class="form-group">
53     @Html.LabelFor(model => model.Telefone, htmlAttributes: new { @class = "control-label col-md-2" })
54     <div class="col-md-10">
55         @Html.EditorFor(model => model.Telefone, new { htmlAttributes = new { @class = "form-control" } })
56         @Html.ValidationMessageFor(model => model.Telefone, "", new { @class = "text-danger" })
57     </div>
58 </div>
59
60 <div class="form-group">
61     @Html.LabelFor(model => model.Logradouro, htmlAttributes: new { @class = "control-label col-md-2" })
62     <div class="col-md-10">
63         @Html.EditorFor(model => model.Logradouro, new { htmlAttributes = new { @class = "form-control" } })
64         @Html.ValidationMessageFor(model => model.Logradouro, "", new { @class = "text-danger" })
65     </div>
66 </div>

```

Fonte: Autoria Própria



### Código 11- Editar Usuário 3/4

```

67
68 <div class="form-group">
69     @Html.LabelFor(model => model.Numero, htmlAttributes: new { @class = "control-label col-md-2" })
70     <div class="col-md-10">
71         @Html.EditorFor(model => model.Numero, new { htmlAttributes = new { @class = "form-control" } })
72         @Html.ValidationMessageFor(model => model.Numero, "", new { @class = "text-danger" })
73     </div>
74 </div>
75
76 <div class="form-group">
77     @Html.LabelFor(model => model.Bairro, htmlAttributes: new { @class = "control-label col-md-2" })
78     <div class="col-md-10">
79         @Html.EditorFor(model => model.Bairro, new { htmlAttributes = new { @class = "form-control" } })
80         @Html.ValidationMessageFor(model => model.Bairro, "", new { @class = "text-danger" })
81     </div>
82 </div>
83
84 <div class="form-group">
85     @Html.LabelFor(model => model.CEP, htmlAttributes: new { @class = "control-label col-md-2" })
86     <div class="col-md-10">
87         @Html.EditorFor(model => model.CEP, new { htmlAttributes = new { @class = "form-control" } })
88         @Html.ValidationMessageFor(model => model.CEP, "", new { @class = "text-danger" })
89     </div>
90 </div>
91
92 <div class="form-group">
93     @Html.LabelFor(model => model.Cidade, htmlAttributes: new { @class = "control-label col-md-2" })
94     <div class="col-md-10">
95         @Html.EditorFor(model => model.Cidade, new { htmlAttributes = new { @class = "form-control" } })
96         @Html.ValidationMessageFor(model => model.Cidade, "", new { @class = "text-danger" })
97     </div>
98 </div>
99

```

Fonte: Autoria Própria

### Código 12 - Editar Usuário 4/4

```

97
98 </div>
99 </div>
100
101 <div class="form-group">
102     @Html.LabelFor(model => model.UF, htmlAttributes: new { @class = "control-label col-md-2" })
103     <div class="col-md-10">
104         @Html.EditorFor(model => model.UF, new { htmlAttributes = new { @class = "form-control" } })
105         @Html.ValidationMessageFor(model => model.UF, "", new { @class = "text-danger" })
106     </div>
107 </div>
108
109 <div class="form-group">
110     <div class="col-md-offset-2 col-md-10">
111         <input type="submit" value="Salvar" class="btn btn-default" />
112     </div>
113 </div>
114
115
116 <div>
117     @Html.ActionLink("Back to List", "Index")
118 </div>
119
120 @section Scripts {
121     @Scripts.Render("~/bundles/jqueryval")
122 }
123

```

Fonte: Autoria Própria

### Código 13 - Exibir Detalhes 1/4

```

Details.cshtml  + X
1  @model PIM.Models.Table
2
3  @{
4      ViewBag.Title = "Details";
5  }
6
7  <h2>Detalhes</h2>
8
9  <div>
10     <h4>Detalhes do Usuário</h4>
11     <hr />
12     <dl class="dl-horizontal">
13         <dt>
14             @Html.DisplayNameFor(model => model.Nome)
15         </dt>
16
17         <dd>
18             @Html.DisplayFor(model => model.Nome)
19         </dd>
20
21         <dt>
22             @Html.DisplayNameFor(model => model.CPF)
23         </dt>
24
25         <dd>
26             @Html.DisplayFor(model => model.CPF)
27         </dd>
28
29         <dt>
30             @Html.DisplayNameFor(model => model.Tipo)
31         </dt>
32
33         <dd>

```

Fonte: Autoria Própria

### Código 14 - Exibir Detalhes 2/4

```

Details.cshtml  + X
34     @Html.DisplayFor(model => model.Tipo)
35     </dd>
36
37     <dt>
38         @Html.DisplayNameFor(model => model.DDD)
39     </dt>
40
41     <dd>
42         @Html.DisplayFor(model => model.DDD)
43     </dd>
44
45     <dt>
46         @Html.DisplayNameFor(model => model.Telefone)
47     </dt>
48
49     <dd>
50         @Html.DisplayFor(model => model.Telefone)
51     </dd>
52
53     <dt>
54         @Html.DisplayNameFor(model => model.Logradouro)
55     </dt>
56
57     <dd>
58         @Html.DisplayFor(model => model.Logradouro)
59     </dd>
60
61     <dt>
62         @Html.DisplayNameFor(model => model.Numero)
63     </dt>
64
65     <dd>
66         @Html.DisplayFor(model => model.Numero)

```

Fonte: Autoria Própria

**Código 15 - Exibir Detalhes 3/4**

```
Details.cshtml  + X
67      </dd>
68
69      <dt>
70          @Html.DisplayNameFor(model => model.Bairro)
71      </dt>
72
73      <dd>
74          @Html.DisplayFor(model => model.Bairro)
75      </dd>
76
77      <dt>
78          @Html.DisplayNameFor(model => model.CEP)
79      </dt>
80
81      <dd>
82          @Html.DisplayFor(model => model.CEP)
83      </dd>
84
85      <dt>
86          @Html.DisplayNameFor(model => model.Cidade)
87      </dt>
88
89      <dd>
90          @Html.DisplayFor(model => model.Cidade)
91      </dd>
92
93      <dt>
94          @Html.DisplayNameFor(model => model.UF)
95      </dt>
96
97      <dd>
98          @Html.DisplayFor(model => model.UF)
99      </dd>
```

Fonte: Autoria Própria

**Código 16 - Exibir Detalhes 4/4**

```
Details.cshtml  + X
97      <dd>
98          @Html.DisplayFor(model => model.UF)
99      </dd>
100
101      </dl>
102  </div>
103  <p>
104      @Html.ActionLink("Editar", "Edit", new { id = Model.Id }) |
105      @Html.ActionLink("Voltar para a Lista", "Index")
106  </p>
107
```

Fonte: Autoria Própria

### Código 17 - Deletar Usuário 1/4

```

Delete.cshtml  [X]
1  @model PIM.Models.Table
2
3  ViewBag.Title = "Delete";
4
5  <h2>Deletar</h2>
6
7  <h3>Tem certeza que deseja deletar esse usuário?</h3>
8
9  <div>
10     <h4>Detalhes do Usuário</h4>
11     <hr />
12     <dl class="dl-horizontal">
13         <dt>
14             @Html.DisplayNameFor(model => model.Nome)
15         </dt>
16         <dd>
17             @Html.DisplayFor(model => model.Nome)
18         </dd>
19         <dt>
20             @Html.DisplayNameFor(model => model.CPF)
21         </dt>
22         <dd>
23             @Html.DisplayFor(model => model.CPF)
24         </dd>
25         <dt>
26             @Html.DisplayNameFor(model => model.Tipo)
27         </dt>
28         <dd>
29             @Html.DisplayFor(model => model.Tipo)
30         </dd>
31     </dl>
32
33

```

Fonte: Autoria Própria

### Código 18 - Deletar Usuário 2/4

```

Delete.cshtml  [X]
34
35     @Html.DisplayFor(model => model.Tipo)
36 </dd>
37
38     <dt>
39         @Html.DisplayNameFor(model => model.DDD)
40     </dt>
41
42     <dd>
43         @Html.DisplayFor(model => model.DDD)
44     </dd>
45
46     <dt>
47         @Html.DisplayNameFor(model => model.Telefone)
48     </dt>
49
50     <dd>
51         @Html.DisplayFor(model => model.Telefone)
52     </dd>
53
54     <dt>
55         @Html.DisplayNameFor(model => model.Logradouro)
56     </dt>
57
58     <dd>
59         @Html.DisplayFor(model => model.Logradouro)
60     </dd>
61
62     <dt>
63         @Html.DisplayNameFor(model => model.Numero)
64     </dt>
65
66     <dd>

```

Fonte: Autoria Própria

### Código 19 - Deletar Usuário 4/4

```
Delete.cshtml  + X
67      @Html.DisplayFor(model => model.Numero)
68      </dd>
69
70      <dt>
71      @Html.DisplayNameFor(model => model.Bairro)
72      </dt>
73
74      <dd>
75      @Html.DisplayFor(model => model.Bairro)
76      </dd>
77
78      <dt>
79      @Html.DisplayNameFor(model => model.CEP)
80      </dt>
81
82      <dd>
83      @Html.DisplayFor(model => model.CEP)
84      </dd>
85
86      <dt>
87      @Html.DisplayNameFor(model => model.Cidade)
88      </dt>
89
90      <dd>
91      @Html.DisplayFor(model => model.Cidade)
92      </dd>
93
94      <dt>
95      @Html.DisplayNameFor(model => model.UF)
96      </dt>
97
98      <dd>
99      @Html.DisplayFor(model => model.UF)
```

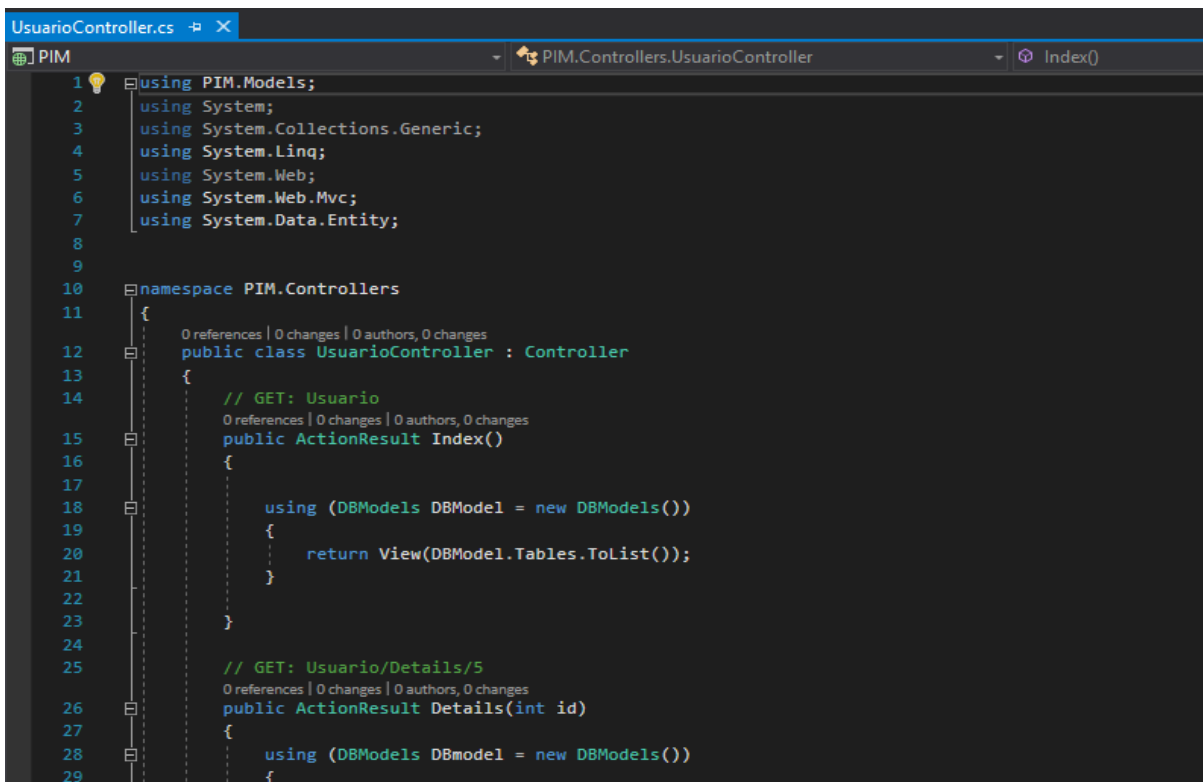
Fonte: Autoria Própria

### Código 20 - Deletar Usuário 4/4

```
Delete.cshtml  + X
97
98      <dd>
99      @Html.DisplayFor(model => model.UF)
100     </dd>
101
102     </dl>
103
104     @using (Html.BeginForm()) {
105         @Html.AntiForgeryToken()
106
107         <div class="form-actions no-color">
108             <input type="submit" value="Deletar" class="btn btn-default" /> |
109             @Html.ActionLink("Voltar para a Lista", "Index")
110         </div>
111     }
112 </div>
113
```

Fonte: Autoria Própria

## Código 21 – Usuário Controller 1/5



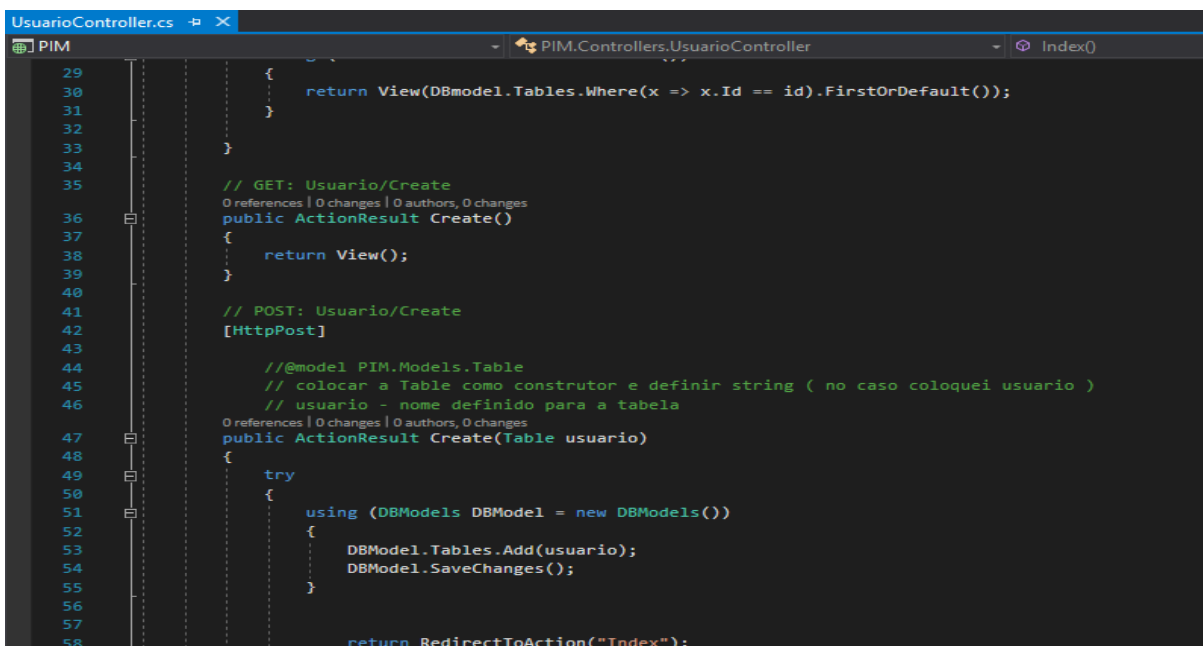
```

1  using PIM.Models;
2  using System;
3  using System.Collections.Generic;
4  using System.Linq;
5  using System.Web;
6  using System.Web.Mvc;
7  using System.Data.Entity;
8
9
10 namespace PIM.Controllers
11 {
12     public class UsuarioController : Controller
13     {
14         // GET: Usuario
15         public ActionResult Index()
16         {
17             using (DBModels DBModel = new DBModels())
18             {
19                 return View(DBModel.Tables.ToList());
20             }
21         }
22
23         // GET: Usuario/Details/5
24         public ActionResult Details(int id)
25         {
26             using (DBModels DBModel = new DBModels())
27             {
28
29

```

Fonte: Autoria Própria

## Código 22 - Usuário Controller 2/5



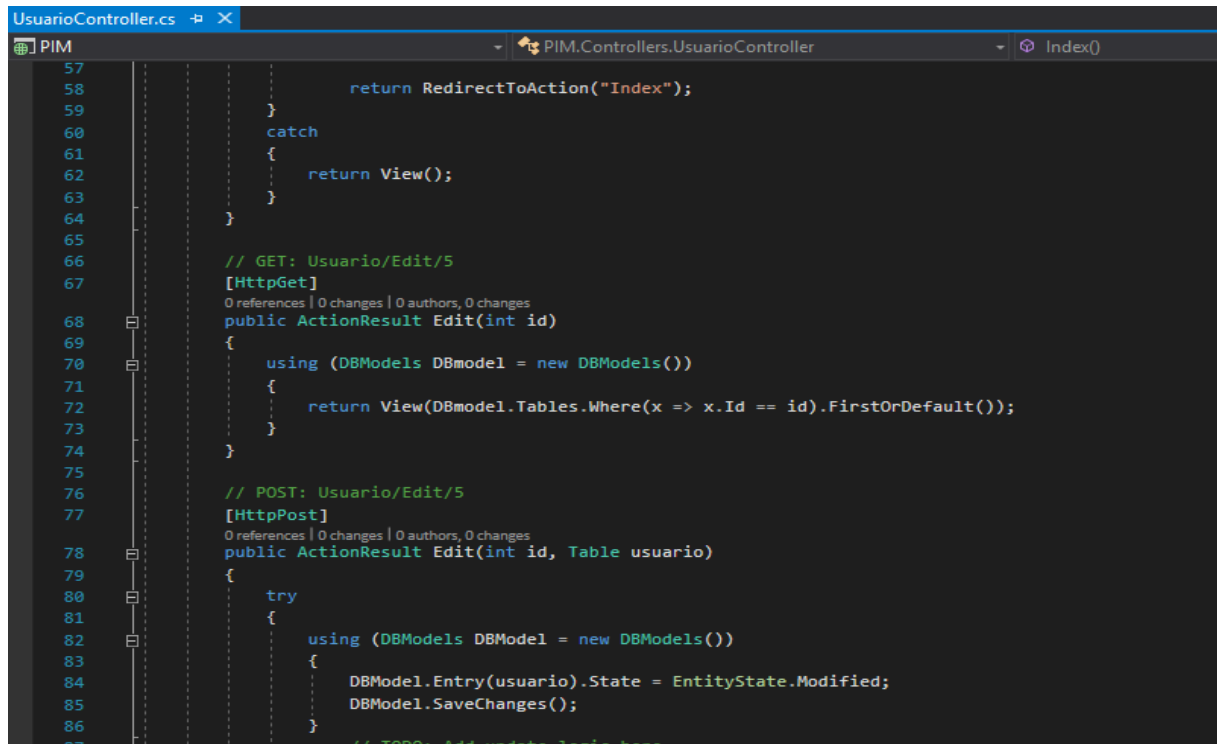
```

29         {
30             return View(DBModel.Tables.Where(x => x.Id == id).FirstOrDefault());
31         }
32     }
33
34     // GET: Usuario/Create
35     public ActionResult Create()
36     {
37         return View();
38     }
39
40     // POST: Usuario/Create
41     [HttpPost]
42     // @model PIM.Models.Table
43     // colocar a Table como construtor e definir string ( no caso coloquei usuario )
44     // usuario - nome definido para a tabela
45     public ActionResult Create(Table usuario)
46     {
47         try
48         {
49             using (DBModels DBModel = new DBModels())
50             {
51                 DBModel.Tables.Add(usuario);
52                 DBModel.SaveChanges();
53             }
54         }
55         catch { }
56     }
57
58     return RedirectToAction("Index");

```

Fonte: Autoria Própria

### Código 23 - Usuário Controller 3/5



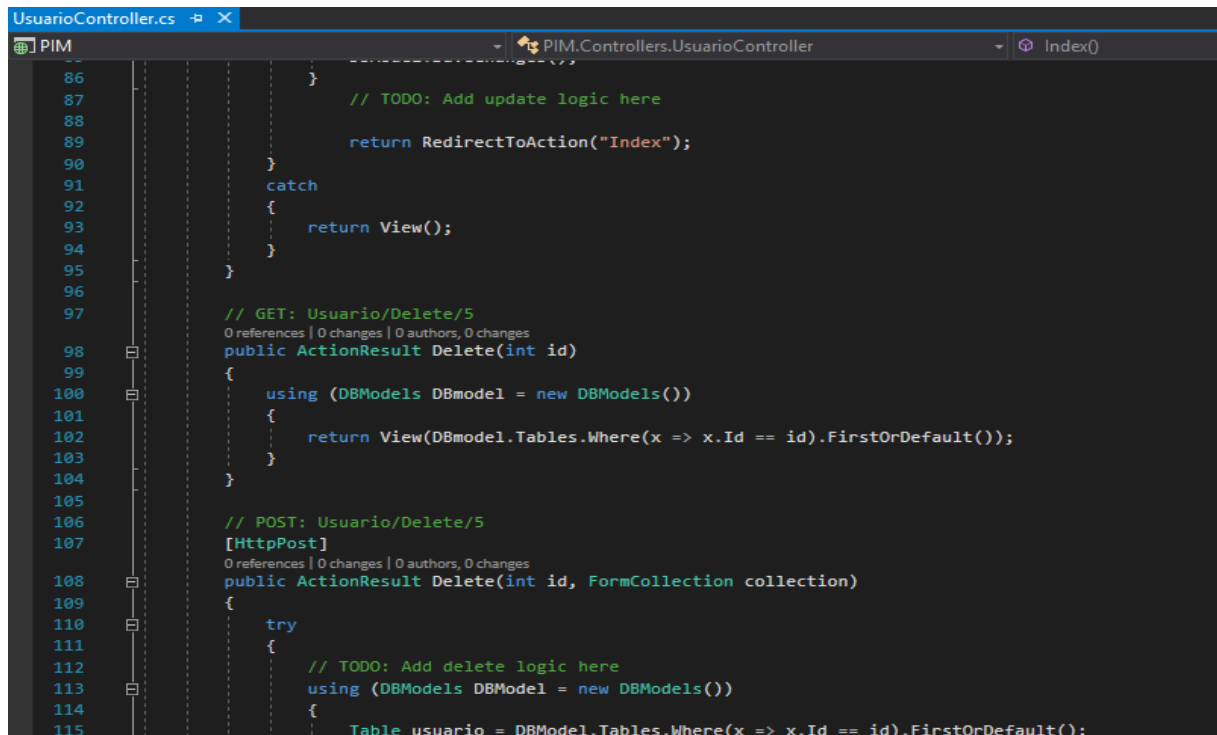
```

57
58         return RedirectToAction("Index");
59     }
60     catch
61     {
62         return View();
63     }
64 }
65
66 // GET: Usuario/Edit/5
67 [HttpGet]
68 0 references | 0 changes | 0 authors, 0 changes
69 public ActionResult Edit(int id)
70 {
71     using (DBModels DBmodel = new DBModels())
72     {
73         return View(DBmodel.Tables.Where(x => x.Id == id).FirstOrDefault());
74     }
75 }
76
77 // POST: Usuario/Edit/5
78 [HttpPost]
79 0 references | 0 changes | 0 authors, 0 changes
80 public ActionResult Edit(int id, Table usuario)
81 {
82     try
83     {
84         using (DBModels DBModel = new DBModels())
85         {
86             DBModel.Entry(usuario).State = EntityState.Modified;
87             DBModel.SaveChanges();
88         }
89         // TODO: Add update logic here
90     }
91     catch
92     {
93         return View();
94     }
95 }
96
97 // GET: Usuario/Delete/5
98 0 references | 0 changes | 0 authors, 0 changes
99 public ActionResult Delete(int id)
100 {
101     using (DBModels DBmodel = new DBModels())
102     {
103         return View(DBmodel.Tables.Where(x => x.Id == id).FirstOrDefault());
104     }
105 }
106
107 // POST: Usuario/Delete/5
108 [HttpPost]
109 0 references | 0 changes | 0 authors, 0 changes
110 public ActionResult Delete(int id, FormCollection collection)
111 {
112     try
113     {
114         // TODO: Add delete logic here
115         using (DBModels DBModel = new DBModels())
116         {
117             Table usuario = DBModel.Tables.Where(x => x.Id == id).FirstOrDefault();
118         }
119     }
120     catch
121     {
122         return View();
123     }
124 }
125

```

Fonte: Autoria Própria

### Código 24 - Usuário Controller 4/5



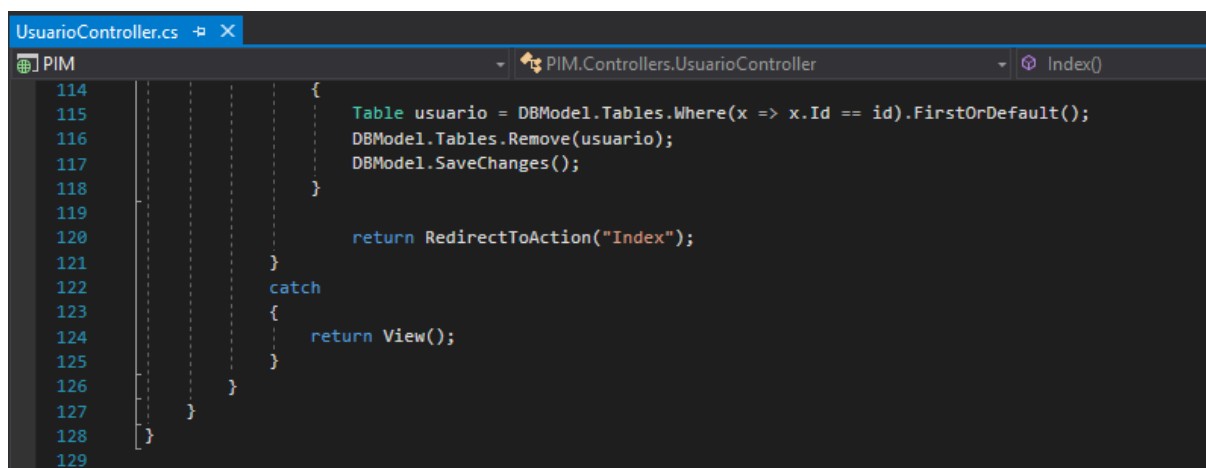
```

86
87     }
88     // TODO: Add update logic here
89     return RedirectToAction("Index");
90 }
91 catch
92 {
93     return View();
94 }
95 }
96
97 // GET: Usuario/Delete/5
98 0 references | 0 changes | 0 authors, 0 changes
99 public ActionResult Delete(int id)
100 {
101     using (DBModels DBmodel = new DBModels())
102     {
103         return View(DBmodel.Tables.Where(x => x.Id == id).FirstOrDefault());
104     }
105 }
106
107 // POST: Usuario/Delete/5
108 [HttpPost]
109 0 references | 0 changes | 0 authors, 0 changes
110 public ActionResult Delete(int id, FormCollection collection)
111 {
112     try
113     {
114         // TODO: Add delete logic here
115         using (DBModels DBModel = new DBModels())
116         {
117             Table usuario = DBModel.Tables.Where(x => x.Id == id).FirstOrDefault();
118         }
119     }
120     catch
121     {
122         return View();
123     }
124 }
125

```

Fonte: Autoria Própria

## Código 25 - Usuário Controller 5/5



```
114         {  
115             Table usuario = DBModel.Tables.Where(x => x.Id == id).FirstOrDefault();  
116             DBModel.Tables.Remove(usuario);  
117             DBModel.SaveChanges();  
118         }  
119  
120         return RedirectToAction("Index");  
121     }  
122     catch  
123     {  
124         return View();  
125     }  
126 }  
127 }  
128 }  
129 }
```

Fonte: Autoria Própria



## 12. Código Fonte do Protótipo de Interface *Mobile*

O código fonte referente aos protótipos de interface *mobile* são no formato .XML, e desenvolvidos utilizando a plataforma Android Studio.

Os campos seguem obedecendo o mesmo padrão desenvolvido para plataforma *web*, tais como campos texto com sua tag texto e campo que possuem números com inserção apenas de números.

### Código 26 - Mobile Navegação

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.drawerlayout.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/drawer_layout"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      android:fitsSystemWindows="true"
9      tools:openDrawer="start">
10
11      <include
12          android:id="@+id/app_bar_main"
13          layout="@layout/app_bar_main"
14          android:layout_width="match_parent"
15          android:layout_height="match_parent" />
16
17      <com.google.android.material.navigation.NavigationView
18          android:id="@+id/nav_view"
19          android:layout_width="wrap_content"
20          android:layout_height="match_parent"
21          android:layout_gravity="start"
22          android:fitsSystemWindows="true"
23          app:headerLayout="@layout/nav_header_main"
24          app:menu="@menu/activity_main_drawer" />
25  </androidx.drawerlayout.widget.DrawerLayout>

```

Fonte: Autoria Própria

## Código 27 - Mobile Cadastro 1/6

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      app:layout_behavior="com.google.android.material.appbar.AppBarLayout$Scrolli..."
8      tools:showIn="@layout/app_bar_main">
9
10     <EditText
11         android:id="@+id/editTextTextPersonName3"
12         android:layout_width="380"
13         android:layout_height="41"
14         android:ems="50"
15         android:inputType="textPersonName"
16         android:text="Nome Completo"
17         tools:layout_editor_absoluteX="19dp"
18         tools:layout_editor_absoluteY="129dp" />
19
20     <EditText
21         android:id="@+id/editTextTextPersonName6"
22         android:layout_width="171dp"
23         android:layout_height="44dp"
24         android:ems="03"
25         android:inputType="number"
26         android:text="DDD"

```

Fonte: Autoria Própria

## Código 28 - Mobile Cadastro 2/6

```

28     tools:layout_editor_absoluteY="172dp" />
29
30     <EditText
31         android:id="@+id/editTextPhone"
32         android:layout_width="wrap_content"
33         android:layout_height="wrap_content"
34         android:ems="13"
35         android:inputType="phone"
36         tools:layout_editor_absoluteX="19dp"
37         tools:layout_editor_absoluteY="219dp"
38         tools:text="Telefone" />
39
40     <EditText
41         android:id="@+id/editTextNumberSigned"
42         android:layout_width="171dp"
43         android:layout_height="44dp"
44         android:ems="13"
45         android:inputType="numberSigned"
46         android:text="CPF"
47         tools:layout_editor_absoluteX="229dp"
48         tools:layout_editor_absoluteY="219dp" />
49
50     <EditText
51         android:id="@+id/editTextTextPostalAddress2"
52         android:layout_width="380dp"
53         android:layout_height="41dp"

```

Fonte: Autoria Própria

**Código 29 - Mobile Cadastro 3/6**

```
52         android:layout_width="380dp"
53         android:layout_height="41dp"
54         android:ems="30"
55         android:inputType="textPostalAddress"
56         android:text="Logradouro"
57         tools:layout_editor_absoluteX="19dp"
58         tools:layout_editor_absoluteY="264dp" />
59
60     <EditText
61         android:id="@+id/editTextTextPersonName8"
62         android:layout_width="wrap_content"
63         android:layout_height="wrap_content"
64         android:ems="20"
65         android:inputType="textPersonName"
66         android:text="Bairro"
67         tools:layout_editor_absoluteX="19dp"
68         tools:layout_editor_absoluteY="305dp" />
69
70     <EditText
71         android:id="@+id/editTextTextPersonName7"
72         android:layout_width="171dp"
73         android:layout_height="44dp"
74         android:ems="7"
75         android:inputType="number"
76         android:text="No."
77         tools:layout_editor_absoluteX="229dp"
```

Fonte: Autoria Própria

### Código 30 - Mobile Cadastro 4/6

```

76         android:text="No."
77         tools:layout_editor_absoluteX="229dp"
78         tools:layout_editor_absoluteY="307dp" />
79
80     <EditText
81         android:id="@+id/editTextTextPostalAddress3"
82         android:layout_width="380dp"
83         android:layout_height="41dp"
84         android:ems="20"
85         android:inputType="textPostalAddress"
86         android:text="Cidade"
87         tools:layout_editor_absoluteX="19dp"
88         tools:layout_editor_absoluteY="351dp" />
89
90     <EditText
91         android:id="@+id/editTextTextPersonName9"
92         android:layout_width="171dp"
93         android:layout_height="44dp"
94         android:ems="2"
95         android:inputType="textPersonName"
96         android:text="UF"
97         tools:layout_editor_absoluteX="229dp"
98         tools:layout_editor_absoluteY="394dp" />
99
100    <EditText
101        android:id="@+id/editTextTextPersonName10"

```

Fonte: Autoria Própria

### Código 31 - Mobile Cadastro 5/6

```

100    <EditText
101        android:id="@+id/editTextTextPersonName10"
102        android:layout_width="wrap_content"
103        android:layout_height="wrap_content"
104        android:ems="9"
105        android:inputType="numberSigned"
106        android:text="CEP"
107        tools:layout_editor_absoluteX="19dp"
108        tools:layout_editor_absoluteY="392dp" />
109
110    <EditText
111        android:id="@+id/editTextTextPersonName5"
112        android:layout_width="wrap_content"
113        android:layout_height="wrap_content"
114        android:ems="10"
115        android:inputType="textPersonName"
116        android:text="Tipo"
117        tools:layout_editor_absoluteX="19dp"
118        tools:layout_editor_absoluteY="170dp" />
119
120    <Button
121        android:id="@+id/button"
122        android:layout_width="wrap_content"
123        android:layout_height="wrap_content"
124        android:text="Salvar"
125        tools:layout_editor_absoluteX="304dp"

```

Fonte: Autoria Própria

### Código 32 - Mobile Cadastro 6/6

```

118         tools:layout_editor_absoluteY="170dp" />
119
120         <Button
121             android:id="@+id/button"
122             android:layout_width="wrap_content"
123             android:layout_height="wrap_content"
124             android:text="Salvar"
125             tools:layout_editor_absoluteX="304dp"
126             tools:layout_editor_absoluteY="549dp" />
127
128         <TextView
129             android:id="@+id/textView2"
130             android:layout_width="252dp"
131             android:layout_height="45dp"
132             android:text="Novo Cadastro"
133             android:textSize="34sp"
134             android:textStyle="bold"
135             tools:layout_editor_absoluteX="24dp"
136             tools:layout_editor_absoluteY="19dp" />
137
138     </androidx.constraintlayout.widget.ConstraintLayout>

```

Fonte: Autoria Própria

### Código 33 - Mobile Pesquisa 1/5

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent">
7
8      <EditText
9          android:id="@+id/editTextTextPersonName15"
10         android:layout_width="393dp"
11         android:layout_height="41dp"
12         android:ems="10"
13         android:inputType="textPersonName"
14         android:text="Bruno Rodrigues"
15         tools:layout_editor_absoluteX="9dp"
16         tools:layout_editor_absoluteY="562dp" />
17
18     <EditText
19         android:id="@+id/editTextTextPersonName13"
20         android:layout_width="393dp"
21         android:layout_height="41dp"
22         android:ems="10"
23         android:inputType="textPersonName"
24         android:text="Bruno Campos"
25         tools:layout_editor_absoluteX="9dp"
26         tools:layout_editor_absoluteY="460dp" />

```

Fonte: Autoria Própria

### Código 34 - Mobile Pesquisa 2/5

```

25      tools:layout_editor_absoluteX="9dp"
26      tools:layout_editor_absoluteY="460dp" />
27
28      <EditText
29          android:id="@+id/editTextTextPersonName14"
30          android:layout_width="393dp"
31          android:layout_height="41dp"
32          android:ems="10"
33          android:inputType="textPersonName"
34          android:text="Bruno Soares"
35          tools:layout_editor_absoluteX="9dp"
36          tools:layout_editor_absoluteY="510dp" />
37
38      <EditText
39          android:id="@+id/editTextTextPersonName12"
40          android:layout_width="393dp"
41          android:layout_height="41dp"
42          android:ems="10"
43          android:inputType="textPersonName"
44          android:text="Bruno Aurelio"
45          tools:layout_editor_absoluteX="9dp"
46          tools:layout_editor_absoluteY="408dp" />
47
48      <EditText
49          android:id="@+id/editTextTextPersonName2"
50          android:layout_width="345dp"

```

Fonte: Autoria Própria

### Código 35 - Mobile Pesquisa 3/5

```

52      android:ems="10"
53      android:inputType="textPersonName"
54      android:text="CPF"
55      tools:layout_editor_absoluteX="25dp"
56      tools:layout_editor_absoluteY="169dp" />
57
58      <TextView
59          android:id="@+id/textView3"
60          android:layout_width="346dp"
61          android:layout_height="42dp"
62          android:text="Consulta "
63          android:textSize="34sp"
64          android:textStyle="bold"
65          tools:layout_editor_absoluteX="22dp"
66          tools:layout_editor_absoluteY="56dp" />
67
68      <SeekBar
69          android:id="@+id/seekBar2"
70          android:layout_width="348dp"
71          android:layout_height="23dp"
72          tools:layout_editor_absoluteX="31dp"
73          tools:layout_editor_absoluteY="671dp" />
74
75      <androidx.constraintlayout.widget.Barrier
76          android:layout_width="wrap_content"
77          android:layout_height="wrap_content"

```

Fonte: Autoria Própria

## Código 36 - Mobile Pesquisa 4/5

```

79      tools:layout_editor_absoluteX="284dp"
80      tools:layout_editor_absoluteY="269dp" />
81
82      <EditText
83          android:id="@+id/editTextTextPersonName"
84          android:layout_width="345dp"
85          android:layout_height="51dp"
86          android:ems="10"
87          android:inputType="textPersonName"
88          android:text="Bruno"
89          tools:layout_editor_absoluteX="25dp"
90          tools:layout_editor_absoluteY="112dp" />
91
92      <EditText
93          android:id="@+id/editTextTextPersonName4"
94          android:layout_width="393dp"
95          android:layout_height="41dp"
96          android:ems="10"
97          android:inputType="textPersonName"
98          android:text="Bruno Aurelio"
99          tools:layout_editor_absoluteX="9dp"
100         tools:layout_editor_absoluteY="359dp" />
101
102      <Button
103          android:id="@+id/button2"
104          android:layout_width="wrap_content"

```

Fonte: Autoria Própria

## Código 37 - Mobile Pesquisa 5/5

```

96      android:ems="10"
97      android:inputType="textPersonName"
98      android:text="Bruno Aurelio"
99      tools:layout_editor_absoluteX="9dp"
100     tools:layout_editor_absoluteY="359dp" />
101
102     <Button
103         android:id="@+id/button2"
104         android:layout_width="wrap_content"
105         android:layout_height="wrap_content"
106         android:text="Pesquisar"
107         tools:layout_editor_absoluteX="252dp"
108         tools:layout_editor_absoluteY="246dp" />
109
110     <SearchView
111         android:layout_width="393dp"
112         android:layout_height="43dp"
113         tools:layout_editor_absoluteX="9dp"
114         tools:layout_editor_absoluteY="306dp" />
115
116 </androidx.constraintlayout.widget.ConstraintLayout>

```

Fonte: Autoria Própria





## REFERÊNCIAS

UM GUIA PARA INICIANTES NA ÁREA WEB. Diferenças entre Front End e Back End Disponível em: <https://tableless.github.io/iniciantes/manual/obasico/o-que-front-back.html> Acessado em: 13 de abril de 2021.

FAÇA RÁPIDO, SITES RESPONSIVOS COM BOOTSTRAP. Bootstrap Disponível em: <https://getbootstrap.com/> Acessado em 09 de maio de 2021.

PÁGINA OFICIAL MICROSOFT VISUAL STUDIO. Microsoft Visual Studio Disponível em: <https://visualstudio.microsoft.com/pt-br/> Acessado em 10 de maio de 2021.

PÁGINA OFICIAL ANDROID STUDIO. Android Studio Disponível em: <https://developer.android.com/studio/> Acessado em 27 de maio de 2021.

AMBIENTE DE DESENVOLVIMENTO INTEGRADO. O que é uma IDE Disponível em: <https://www.redhat.com/pt-br/topics/middleware/what-is-ide> Acessado em 10 de maio de 2021.

PÁGINA OFICIAL BANCO DE DADOS MYSQL Disponível em: <https://www.mysql.com/downloads/> Acessado em 12 de maio de 2021

O QUE É E COMO USAR O MYSQL Disponível em: <https://www.techtudo.com.br/artigos/noticia/2012/04/o-que-e-e-como-usar-o-mysql.html> Acessado em 12 de maio de 2021

DAVENPORT, Thomas H.; PRUSAK, Laurence. Conhecimento Empresarial; como as organizações gerenciam o seu capital intelectual. Rio de Janeiro: Campus, 1998.

GUNJI, José Cassiano, MARINHO, Salatiel Luz, SOUZA, Luciano Soares de. Tópicos Especiais de Programação Orientada a Objetos. São Paulo: UNIP Interativa, 2019.

MARINHO, Salatiel Luz, SOUZA, Luciano Soares de. Programação Orientada a Objetos II. São Paulo: UNIP Interativa, 2019.