

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS  
ESCOLA DE CIÊNCIAS EXATAS E DA COMPUTAÇÃO  
GRADUAÇÃO EM ENGENHARIA DE COMPUTAÇÃO



**DESENVOLVIMENTO DE UM APLICATIVO PARA GERENCIAMENTO DE  
PEDIDOS DELIVERY DE UM RESTAURANTE**

LEONARDO MORAES REBELATTO

GOIÂNIA  
2020

LEONARDO MORAES REBELATTO

**DESENVOLVIMENTO DE UM APLICATIVO PARA GERENCIAMENTO DE  
PEDIDOS DELIVERY DE UM RESTAURANTE**

Trabalho de conclusão de Curso apresentado à Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, como parte dos requisitos para a obtenção do título de Bacharel em Engenharia de Computação.

Orientador(a):

Profa. Ma. Ludmilla Reis Pinheiro dos Santos

Banca examinadora:

Prof. Me. Geraldo Valeriano Ribeiro

Prof. Dr. José Luiz de Freitas Júnior

GOIÂNIA  
2020

LEONARDO MORAES REBELATTO

**DESENVOLVIMENTO DE UM APLICATIVO PARA GERENCIAMENTO DE  
PEDIDOS DELIVERY DE UM RESTAURANTE**

Trabalho de Conclusão de Curso aprovado em sua forma final pela Escola de Ciências Exatas e da Computação, da Pontifícia Universidade Católica de Goiás, para obtenção do título de Bacharel em Engenharia de Computação, em \_\_\_\_/\_\_\_\_/\_\_\_\_\_.

---

Orientadora: Profa. Ma. Ludmilla Reis Pinheiro dos Santos

---

Profa. Ma. Ludmilla Reis Pinheiro dos Santos  
Coordenadora de Trabalho de Conclusão de Curso

GOIÂNIA  
2020

## **AGRADECIMENTOS**

Agradeço em primeiro lugar a minha família por me auxiliarem e me apoiarem sempre em toda minha trajetória.

Agradeço a minha companheira e namorada, Gabriela Quevedo, por todo este tempo estar do meu lado me apoiando.

Agradeço à professora Ludmilla, por ter me orientado durante todo o processo deste trabalho, que mesmo com todas as dificuldades neste ano de 2020, sempre me guiou e ajudou a superar as barreiras.

E por fim agradeço a todos os professores do curso de Engenharia de Computação da PUC Goiás que eu tive a honra de conhecer e que puderam passar o conhecimento me permitindo chegar até este momento.

## RESUMO

Os aplicativos têm inovado a cada dia o mercado, seja ele alimentício, tecnológico, de saúde e vários outros. Se tornando uma parte importante na vida de pessoas, os aplicativos procuram trazer conforto, facilidade de acesso, dentre outras diversas melhorias na vida dos usuários. Dito isto, este trabalho apresenta o levantamento de requisitos, a documentação e a implementação de um aplicativo para restaurante, denominado O Comilão, utilizando o *framework* Flutter e o banco de dados Firebase. O aplicativo tem o objetivo de trazer um cardápio digital para que em tempos de pandemia, como o que vivemos hoje, os clientes possam ter mais segurança ao escolher o produto que deseja consumir e a facilidade de usar seu próprio *smartphone*. O aplicativo também apresenta a opção de *delivery* para os clientes que querem aproveitar a refeição no conforto de sua residência, por exemplo. O aplicativo pode ser utilizado de três formas: (a) com usuário sem efetuar login, apenas para visualização do cardápio; (b) com o usuário logado, para fazer seu pedido *delivery*; e (c) com o usuário logado como administrador, podendo gerenciar os pedidos do restaurante. As informações dos produtos contidos no cardápio e também as imagens estarão armazenadas em nuvem no Firebase.

***Palavras-Chave:*** *aplicativo, firebase, restaurante, delivery.*

## ABSTRACT

The applications have been innovating the market every day, be it food, technology, health and several others. Becoming an important part of people's lives, applications seek to bring comfort, ease of access, among other diverse improvements in the lives of users. That said, this paper presents the requirements survey, the documentation and the implementation of an application for a restaurant, called O Comilão, using the Flutter framework and the Firebase database. The application aims to bring a digital menu so that in times of pandemic, like the one we live in today, customers can have more security when choosing the product they want to consume and the ease of using their own smartphone. The application also presents the delivery option for customers who want to enjoy the meal in the comfort of their home, for example. The application can be used in three ways: (a) with a user without logging in, just to view the menu; (b) with the logged in user, to place their delivery order; and (c) with the user logged in as an administrator, being able to manage restaurant orders. The product information contained in the menu and also the images will be stored in the cloud in Firebase.

***Key-Words:*** *application, firebase, restaurant, delivery.*

## LISTA DE IMAGENS

Imagem 1 - Diagrama de casos de uso. ....	24
Imagem 2 - Widgets. ....	33
Imagem 3 - Cloud Firestore.....	34
Imagem 4 - Usuários do aplicativo.....	34
Imagem 5 - Template redefinição de senha.....	35
Imagem 6 - Métodos de Login. ....	35
Imagem 7 - Visual Studio Code. ....	36
Imagem 8 - Tela Home. ....	38
Imagem 9 - Tela Home posição pratos. ....	39
Imagem 10 - Tela produto. ....	40
Imagem 11 - Tela de login.....	41
Imagem 12 - Tela de cadastro.....	42
Imagem 13 - Tela de recuperação de senha.....	43
Imagem 14 - Menu lateral com usuário sem efetuar login. ....	44
Imagem 15 - Menu lateral com usuário logado. ....	45
Imagem 16 - Tela carrinho. ....	46
Imagem 17 - Tela pedidos. ....	47
Imagem 18 - Tela Perfil.....	48
Imagem 19 - Menu lateral com usuário admin.....	49
Imagem 20 - Tela gerenciamento de pedidos.....	50

## LISTA DE QUADROS

Quadro 1 - RF 001 Manter Usuário.....	17
Quadro 2 - RF 002 Realizar login. ....	17
Quadro 3 - RF 003 Recuperar senha. ....	18
Quadro 4 - RF 004 Visualizar Cardápio.....	18
Quadro 5 - RF 005 Realizar pedido.....	19
Quadro 6 - RF 006 Visualizar pedidos. ....	19
Quadro 7 - RF 007 Adicionar endereço.....	20
Quadro 8 - RF 008 Gerenciar pedidos.....	20
Quadro 9 - RQ 001 Confidencialidade. ....	21
Quadro 10 - RQ 002 Usabilidade. ....	21
Quadro 11 - RQ 003 Confiabilidade.....	21
Quadro 12 - RQ 004 Desempenho. ....	22
Quadro 13 - RQ 005 Manutenibilidade. ....	22
Quadro 14 – DD 001 Dados do usuário ....	22
Quadro 15 – DD 002 Dados do produto.....	23
Quadro 16 – DD 003 Dados do pedido. ....	23
Quadro 17 - CSU 001 Visualizar cardápio.....	25
Quadro 18 - CSU 002 Realizar login.....	26
Quadro 19 - CSU 003 Cadastrar Usuário. ....	27
Quadro 20 - CSU 004 Recuperar senha.....	27
Quadro 21 - CSU 005 Realizar pedido.....	28
Quadro 22 - CSU 006 Visualizar pedidos. ....	29
Quadro 23 - CSU 007 Visualizar perfil.....	30
Quadro 24 - CSU 008 Gerenciar pedidos.....	30

## SUMÁRIO

<b>1 INTRODUÇÃO</b> .....	<b>11</b>
1.1 Objetivo Geral.....	12
1.2 Objetivos Específicos .....	12
1.3 Justificativa .....	12
1.4 Stakeholders .....	12
<b>2 DESCRIÇÃO GLOBAL</b> .....	<b>13</b>
2.1 Aspecto Geral do Produto.....	13
2.1.1 Interfaces do Sistema.....	13
2.1.2 Interfaces do Usuário .....	14
2.1.3 Interfaces de hardware .....	14
2.1.4 Interfaces de Software.....	14
2.1.5 Interfaces de Comunicação.....	14
2.2 Funções do Sistema.....	14
2.3 Restrições/Limitações .....	15
2.4 Regras de negócio .....	15
<b>3 REQUISITOS ESPECÍFICOS</b> .....	<b>16</b>
3.1 Requisitos de Usuário - Necessidades .....	16
3.2 Requisitos Funcionais.....	17
3.3 Requisitos de Qualidade.....	21
3.4 Dicionário de Dados.....	22
<b>4 CASOS DE USO</b> .....	<b>24</b>
4.1 Diagrama de casos de Uso.....	24
4.2 Modelo Descritivo de casos de uso .....	25
<b>5 DESENVOLVIMENTO DO APLICATIVO</b> .....	<b>32</b>
5.1 Dart .....	32
5.2 Flutter .....	32
5.2.1 <i>Hot-Reload</i> .....	32
5.2.2 <i>Widgets</i> .....	32
5.3 Firebase.....	33
5.4 Visual Studio Code .....	36
<b>6 IMPLEMENTAÇÕES E RESULTADOS</b> .....	<b>37</b>
6.1 Telas Compartilhadas .....	37

<b>6.1.1 Tela Home .....</b>	<b>37</b>
<b>6.1.2 Tela produto.....</b>	<b>39</b>
<b>6.1.3 Tela de login .....</b>	<b>40</b>
<b>6.1.4 Tela de cadastro .....</b>	<b>41</b>
<b>6.1.5 Tela de recuperação de senha.....</b>	<b>42</b>
<b>6.2 Telas com usuário sem efetuar login.....</b>	<b>43</b>
<b>6.2.1 Menu lateral com usuário sem efetuar login.....</b>	<b>43</b>
<b>6.3 Telas com cliente logado.....</b>	<b>44</b>
<b>6.3.1 Menu lateral com usuário logado.....</b>	<b>44</b>
<b>6.3.2 Tela carrinho.....</b>	<b>45</b>
<b>6.3.3 Tela pedidos.....</b>	<b>46</b>
<b>6.3.4 Tela perfil .....</b>	<b>47</b>
<b>6.4 Telas com usuário logado como administrador .....</b>	<b>48</b>
<b>6.4.1 Menu lateral com usuário logado como administrador .....</b>	<b>48</b>
<b>6.4.2 Tela Gerenciamento de pedidos .....</b>	<b>49</b>
<b>7 CONSIDERAÇÕES FINAIS.....</b>	<b>51</b>
<b>7.1 Trabalhos Futuros .....</b>	<b>51</b>
<b>REFERÊNCIAS.....</b>	<b>52</b>
<b>APÊNDICE A: BRIEFING DO SISTEMA .....</b>	<b>54</b>
<b>ANEXO A – TERMO DE AUTORIZAÇÃO DE PUBLICAÇÃO DE PRODUÇÃO ACADÊMICA.....</b>	<b>55</b>

## 1 INTRODUÇÃO

O uso de *smartphones* para acesso à internet já faz parte da vida da maioria dos brasileiros, uma simples pesquisa em páginas de busca fornece a maioria das informações que precisa, sendo que a maioria dos casos essas pesquisas são realizadas a partir de *smartphones*. Segundo o Instituto Brasileiro de Geografia e Estatística (IBGE), no ano de 2018, 99,2% dos domicílios que havia acesso à internet, o celular era utilizado para este fim (IBGE, 2018).

*“Os dispositivos móveis, em especial, os aplicativos móveis, visam atender o acesso das pessoas à informação e ao conhecimento, sem restrição de tempo e espaço. A possibilidade da queda de barreiras de tempo e espaço permite também novas formas de comunicação. Tais características agregam valor estratégico para a nova sociedade da Era da Informação.” (BARRA et. al.,2017).*

Neste contexto, com a diminuição do tempo livre devido a atividades da vida cotidiana, a utilização de *smartphones* tem auxiliado na procura por aplicativos de *delivery* de comida, que fornecem facilidade e rapidez na compra de alimentação.

Visando uma necessidade da população, o serviço digital alimentício por meio de aplicativos tem aumentado e procura alcançar mais clientes. Segundo a pesquisa *Consumer Reports on Eating Share Trends* (CREST), os pedidos cresceram 71% em 2019 só no setor digital, mostrando vários benefícios e facilidades como: atualização de cardápio e preços, facilidade na escolha do pagamento, ao acrescentar adicionais e ao personalizar o pedido (VEJA, 2020).

Os restaurantes devem se atentar as tendências de serviços digitais. Segundo Chiavenato (2008), "O mundo está mudando com uma rapidez incrível. E com intensidade cada vez maior. A mudança sempre existiu na história da humanidade, mas não com o volume, rapidez e impacto com que ocorre hoje." A concorrência cria a necessidade de aprimoramento, sendo que o serviço de atendimento é um dos serviços que pode ser melhorado com a digitalização (NEDEL, 2020).

O aumento de uso de aplicativos, crescimento no mercado de *delivery* e necessidade de aprimoramento, despertou o interesse em desenvolver uma aplicação para facilitar o setor de entregas de um restaurante, digitalizar o cardápio e automatizar os pedidos para os clientes que estão no local.

## 1.1 Objetivo Geral

Este trabalho tem o objetivo de desenvolver e documentar um aplicativo mobile para restaurante, fornecendo um cardápio digital para clientes que estão no local terem mais informações sobre os produtos sem precisar pedir para o garçom ou pegar o cardápio físico, e ainda a opção de delivery para os clientes que querem aproveitar sua refeição no conforto de sua casa ou onde preferir.

## 1.2 Objetivos Específicos

- Documentação das necessidades dos usuários;
- Documentação dos requisitos funcionais e de qualidade;
- Modelagem do software: diagrama de casos de uso;
- Desenvolvimento do aplicativo utilizando o *Framework* Flutter;
- Análise dos resultados.

## 1.3 Justificativa

Com a pandemia da COVID-19, em 2020, restaurantes e bares tiveram que se reinventar com adaptações e mudanças para prevenir a saúde de clientes e funcionários. Algumas dessas adaptações mesmo em um período pós-pandemia continuarão a existir, pois o isolamento forçado fez as pessoas criarem novos hábitos, ter mais cuidado fora de casa ou se acostumar e se sentir mais confortável pedindo seu prato favorito em casa.

## 1.4 Stakeholders

Os *stakeholders* deste projeto incluem a administração e todos os clientes do restaurante O Comilão, localizado em Caldas Novas-GO, desde os clientes que vão ao local aproveitar sua refeição ou os que preferem ficar no conforto e segurança de suas casas.

## 2 DESCRIÇÃO GLOBAL

Este capítulo apresenta uma descrição do aplicativo, juntamente com as descrições das interfaces, as funções e suas restrições.

### 2.1 Aspecto Geral do Produto

O aplicativo foi construído usando o *framework* Flutter com o propósito de fornecer um cardápio virtual para os clientes do restaurante O Comilão, localizado em Caldas Novas-GO, e ainda com a opção de *delivery* para os clientes que não estão no local.

#### 2.1.1 Interfaces do Sistema

O aplicativo O Comilão possui as seguintes interfaces:

- **Tela de Login:** Contém os campos de e-mail e senha para acesso a conta, opção para recuperar senha, fazer cadastro e entrar com o Facebook ou Google;
- **Tela de Cadastro:** Contém os campos para preenchimento das informações necessárias para o cadastro do usuário;
- **Tela de Recuperação de senha:** Contém os campos para preenchimento das informações necessárias para a recuperação da senha do usuário;
- **Tela de Perfil:** Contém as informações do usuário, tais como nome, telefone, e-mail, foto e a opção para editar o perfil;
- **Home:** Contém o cardápio do restaurante, com todos os produtos disponíveis, botão para acessar o carrinho e menu lateral com dois formatos diferentes. Quando o usuário está logado, apresenta as opções para acessar as telas Home, Pedidos, Perfil e ainda a opção de *logout*. Quando o usuário não está logado, mostra a opção para o usuário fazer login;
- **Menu lateral:** O menu lateral é uma interface que compõe a tela Home com 3 formatos diferentes, quando o usuário não fez login, quando o usuário está logado e quando o usuário está logado como administrador;
- **Tela do Produto:** Contém a descrição do produto, foto, campo para observação e a opção de adicionar ao carrinho;
- **Tela Carrinho:** Contém a lista de produtos com as respectivas informações, valores parciais, valor da entrega e valor total. Opções de aumentar ou diminuir quantidade de cada produto, remover produto, adicionar cupom de desconto, selecionar método de pagamento e endereço, solicitar troco e finalizar compra;

- **Tela de Pedidos:** Contém um histórico de todos os pedidos do usuário com as devidas informações e status;
- **Tela Gerenciamento de Pedidos:** Contém todos os pedidos do restaurante com as devidas informações e botões para mudar os status dos pedidos.

### 2.1.2 Interfaces do Usuário

O sistema será desenvolvido para ser utilizado em *smartphones* ou *tablets* que tenham o sistema operacional Android com acesso à internet.

### 2.1.3 Interfaces de hardware

O software foi desenvolvido para os requisitos mínimos:

Processador: *QuadCore* (1.8GHz);

Memória RAM: 4 GigaBytes;

Armazenamento livre: 100 MegaBytes de armazenamento livre.

### 2.1.4 Interfaces de Software

O sistema requer sistema operacional Android com a versão sendo a Marshmallow 6.0.1 ou mais recente.

### 2.1.5 Interfaces de Comunicação

O sistema utilizará o Firebase para armazenamento de dados dos produtos, pedidos e informações dos usuários, e ainda para os serviços de autenticação de Usuário.

## 2.2 Funções do Sistema

Esta seção apresenta as principais funções que o sistema realizará:

- Cadastrar usuário;
- Realizar login;
- Visualizar cardápio;
- Fazer pedido;
- Selecionar método de pagamento e solicitar troco;
- Visualizar histórico de pedidos com seus devidos status;
- Gerenciar pedidos do restaurante;
- Visualizar e editar informações próprias dos usuários.

### **2.3 Restrições/Limitações**

O aplicativo possui as seguintes restrições:

- O sistema só pode ser acessado quando o usuário tiver conexão com a internet;
- Caso o usuário não estiver logado, conseguirá apenas visualizar os produtos;
- A tela de gerenciamento de pedidos só deverá ser acessada pelo usuário admin;
- O usuário só terá foto de perfil caso tenha efetuado login com o Facebook ou Gmail.

### **2.4 Regras de negócio**

O aplicativo possui a seguinte regra de negócio:

- Se o cupom de desconto inserido pelo usuário for válido, o valor total do pedido deve ter o desconto de acordo com a porcentagem do cupom cadastrada no banco de dados.

### 3 REQUISITOS ESPECÍFICOS

Este capítulo apresenta os Requisitos de Usuário - Necessidades (RUN), os Requisitos Funcionais (RF) e os Requisitos de Qualidade (RQ), definidos a partir do processo de levantamento de requisitos, com auxílio de uma entrevista com os responsáveis pelo restaurante disponível no Apêndice A: *Briefing* do Sistema.

#### 3.1 Requisitos de Usuário - Necessidades

Esta seção apresenta os requisitos de usuário - necessidades definidos no processo de levantamento de requisitos, com auxílio de uma entrevista com os responsáveis pelo restaurante disponível no Apêndice A: *Briefing* do Sistema.

Tabela 1- Requisitos de Usuário – Necessidade.

<b>Id</b>	<b>Descrição</b>	<b>Fonte</b>
RUN 001	O usuário cliente sem efetuar login deve ser capaz de se registrar usando um e-mail válido e uma senha, Facebook ou Gmail.	Autoria própria
RUN 002	O usuário cliente deve ser capaz de realizar login.	Autoria própria
RUN 003	O usuário cliente deve ser capaz de recuperar senha, caso esqueça a senha cadastrada.	Autoria própria
RUN 004	O usuário cliente deve ser capaz de inserir um número de telefone válido em seu cadastro.	<i>Briefing</i>
RUN 005	O usuário cliente sem efetuar login deve ser capaz de visualizar o cardápio sem realizar login.	<i>Briefing</i>
RUN 006	O usuário cliente deve ser capaz de adicionar produtos ao seu carrinho.	Autoria própria
RUN 007	O usuário cliente deve ser capaz de editar seu perfil e escrever o endereço para entrega.	Autoria própria
RUN 008	O usuário cliente deve ser capaz de acessar o histórico de pedidos.	Autoria própria
RUN 009	O usuário cliente deve ser capaz de editar seu perfil.	Autoria própria
RUN 010	O usuário deve ser capaz de ir diretamente para o tipo de produto que deseja na tela do cardápio.	<i>Briefing</i>
RUN 011	O usuário admin deve ser capaz de gerenciar os pedidos do restaurante.	Autoria própria

Fonte: Elaborado pelo autor.

### 3.2 Requisitos Funcionais

Esta seção apresenta os requisitos funcionais definidos no processo de levantamento de requisitos.

Quadro 1 - RF 001 Manter Usuário.

Identificador	Nome
RF 001	Manter Usuário
Caso de Uso	Autor
CSU 003, CSU 008	Leonardo Moraes Rebelatto
Descrição	
O aplicativo deve ter a opção de manter usuário (Criar e editar).	
Critério de Verificação	
<ul style="list-style-type: none"> <li>Os campos devem ser preenchidos;</li> </ul>	
Dependência	Prioridade
RUN 001, RUN 004, DD 001	Essencial

Fonte: Elaborado pelo autor.

Quadro 2 - RF 002 Realizar login.

Identificador	Nome
RF 002	Realizar login
Caso de Uso	Autor
CSU 002	Leonardo Moraes Rebelatto
Descrição	
O login deve ser realizado usando um e-mail ou telefone e senha já cadastrados. O usuário também terá a opção de entrar com Facebook ou Gmail.	
Critério de Verificação	
<ul style="list-style-type: none"> <li>Verificar se o login e senha estão cadastrados.</li> </ul>	
Dependência	Prioridade
RUN 002	Essencial

Fonte: Elaborado pelo autor.

Quadro 3 - RF 003 Recuperar senha.

<b>Identificador</b>	<b>Nome</b>	
RF 003	Recuperar senha	
<b>Caso de Uso</b>		<b>Autor</b>
CSU 004		Leonardo Moraes Rebelatto
<b>Descrição</b>		
Caso o usuário esqueça a senha, a aplicação deve ter uma opção de recuperá-la. O usuário deve informar o e-mail cadastrado no aplicativo para receber um <i>link</i> para a definição de uma nova senha.		
<b>Critério de Verificação</b>		
<ul style="list-style-type: none"> <li>• Verificar se o e-mail está cadastrado.</li> </ul>		
<b>Dependência</b>		<b>Prioridade</b>
RUN 003		Essencial

Fonte: Elaborado pelo autor.

Quadro 4 - RF 004 Visualizar Cardápio.

<b>Identificador</b>	<b>Nome</b>	
RF 004	Visualizar Cardápio	
<b>Caso de Uso</b>		<b>Autor</b>
CSU 001		Leonardo Moraes Rebelatto
<b>Descrição</b>		
<p>O aplicativo deve entrar sem cadastro para que seja possível acessar o cardápio apenas para a visualização.</p> <p>Deve ser possível ir diretamente para o tipo de produto que o usuário deseja no cardápio: sanduíches, pizzas, pratos, bebidas, entre outros.</p>		
<b>Critério de Verificação</b>		
<b>Dependência</b>		<b>Prioridade</b>
RUN 005, RUN 010		Essencial

Fonte: Elaborado pelo autor.

Quadro 5 - RF 005 Realizar pedido.

<b>Identificador</b>	<b>Nome</b>
RF 005	Realizar pedido
<b>Caso de Uso</b>	<b>Autor</b>
CSU 005	Leonardo Moraes Rebelatto
<b>Descrição</b>	
Permite ao cliente realizar o pedido, para isso a aplicação deverá disponibilizar para cada produto a opção de adicionar ao carrinho. Quando o pedido do usuário estiver completo, ele deverá acessar o carrinho e selecionar a opção “Finalizar Pedido”.	
<b>Critério de Verificação</b>	
<ul style="list-style-type: none"> <li>• O usuário deve estar logado;</li> <li>• O carrinho não deve estar vazio;</li> <li>• O endereço deve estar preenchido;</li> <li>• O usuário deve ter um número de telefone válido cadastrado;</li> <li>• O usuário deve selecionar o método de pagamento.</li> </ul>	
<b>Dependência</b>	<b>Prioridade</b>
RUN 006, RUN 007	Essencial

Fonte: Elaborado pelo autor.

Quadro 6 - RF 006 Visualizar pedidos.

<b>Identificador</b>	<b>Nome</b>
RF 006	Visualizar pedidos.
<b>Caso de Uso</b>	<b>Autor</b>
CSU 006	Leonardo Moraes Rebelatto
<b>Descrição</b>	
O aplicativo deve ter um histórico dos pedidos do usuário.	
<b>Critério de Verificação</b>	
<ul style="list-style-type: none"> <li>• O usuário deve estar logado;</li> </ul>	
<b>Dependência</b>	<b>Prioridade</b>
RUN 008	Essencial

Fonte: Elaborado pelo autor.

Quadro 7 - RF 007 Adicionar endereço.

<b>Identificador</b>	<b>Nome</b>	
RF 007	Adicionar endereço	
<b>Caso de Uso</b>		<b>Autor</b>
CSU 004		Leonardo Moraes Rebelatto
<b>Descrição</b>		
O aplicativo deve ter a opção de adicionar o endereço que o cliente deseja receber o pedido.		
<b>Critério de Verificação</b>		
<ul style="list-style-type: none"> <li>Os campos devem estar preenchidos;</li> </ul>		
<b>Dependência</b>		<b>Prioridade</b>
RUN 007		Essencial

Fonte: Elaborado pelo autor.

Quadro 8 - RF 008 Gerenciar pedidos.

<b>Identificador</b>	<b>Nome</b>	
RF 008	Gerenciar pedidos	
<b>Caso de Uso</b>		<b>Autor</b>
CSU 008		Leonardo Moraes Rebelatto
<b>Descrição</b>		
O aplicativo deve ter uma opção de gerenciar pedidos, quando o usuário for um administrador.		
<b>Critério de Verificação</b>		
<ul style="list-style-type: none"> <li>Usuário logado como administrador.</li> </ul>		
<b>Dependência</b>		<b>Prioridade</b>
RUN 011		Essencial

Fonte: Elaborado pelo autor.

### 3.3 Requisitos de Qualidade

Esta seção apresenta os requisitos de qualidade definidos no processo de levantamento de requisitos.

Quadro 9 - RQ 001 Confidencialidade.

Identificador	RQ 001
Nome	Confidencialidade
Caso de Uso	
Descrição	As informações dos usuários são restritas e só podem ser acessadas e visualizadas pelo próprio.
Autor	Leonardo Moraes Rebelatto
Critério de Verificação	
Dependência	
Prioridade	Essencial

Fonte: Elaborado pelo autor.

Quadro 10 - RQ 002 Usabilidade.

Identificador	RQ 002
Nome	Usabilidade
Caso de Uso	
Descrição	Os clientes do restaurante devem conseguir usar o aplicativo sem precisar de ajuda.
Autor	Leonardo Moraes Rebelatto
Critério de Verificação	
Dependência	
Prioridade	Essencial

Fonte: Elaborado pelo autor.

Quadro 11 - RQ 003 Confiabilidade.

Identificador	RQ 003
Nome	Confiabilidade
Caso de Uso	
Descrição	O servidor deve ser confiável podendo suportar vários usuários simultâneos.
Autor	Leonardo Moraes Rebelatto
Critério de Verificação	<ul style="list-style-type: none"> <li>• Teste de estresse de servidor.</li> </ul>
Dependência	
Prioridade	Essencial

Fonte: Elaborado pelo autor.

Quadro 12 - RQ 004 Desempenho.

Identificador	RQ 004
Nome	Desempenho
Caso de Uso	
Descrição	O aplicativo deve ter um bom desempenho, com os botões espaçados, com poucos carregamentos e ser responsivo.
Autor	Leonardo Moraes Rebelatto
Critério de Verificação	<ul style="list-style-type: none"> <li>• O aplicativo não deve demorar mais que 5 segundos para abrir;</li> <li>• O tempo não deve passar de 10 segundos para finalizar o pedido.</li> </ul>
Dependência	
Prioridade	Essencial

Fonte: Elaborado pelo autor.

Quadro 13 - RQ 005 Manutenibilidade.

Identificador	RQ 005
Nome	Manutenibilidade
Caso de Uso	
Descrição	O aplicativo deve ser de fácil manutenção, com os códigos bem escritos e bem comentados.
Autor	Leonardo Moraes Rebelatto
Critério de Verificação	
Dependência	
Prioridade	Essencial

Fonte: Elaborado pelo autor.

### 3.4 Dicionário de Dados

Esta seção apresenta o dicionário de dados, contido nos quadros 14 ao 16, das informações que serão salvas no banco de dados para cada classe presente no aplicativo.

Quadro 14 – DD 001 Dados do usuário

Identificador	Descrição			
DD 001	Dados do usuário			
Identificador	Tamanho	Tipo	Formato	Descrição
nome	30	String		Nome do usuário
email	35	String		E-mail do usuário
telefone	11	String		Telefone do usuário
userId	28	String		Código de identificação de usuário gerado pelo Firebase.

Fonte: Elaborado pelo autor.

Quadro 15 – DD 002 Dados do produto.

<b>Identificador</b>	<b>Descrição</b>			
DD 002	Dados do produto			
<b>Identificador</b>	<b>Tamanho</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descrição</b>
nome	30	String		Nome do produto.
descricao	500	String		Descrição do produto.
foto	100	String		Link da foto do produto.
preco	4	double		Preço do produto.
tipo	16	String		Tipo do produto(sanduiche, pizza, prato, etc.)

Fonte: Elaborado pelo autor.

Quadro 16 – DD 003 Dados do pedido.

<b>Identificador</b>	<b>Descrição</b>			
DD 003	Dados do pedido			
<b>Identificador</b>	<b>Tamanho</b>	<b>Tipo</b>	<b>Formato</b>	<b>Descrição</b>
clienteId	28	String		Código de identificação de usuário gerado pelo Firebase.
desconto	10	String		Valor desconto inserido pelo código do cliente.
precoEntrega	4	double		Preço da entrega do pedido.
precoProdutos	5	double		Preço da soma dos produtos.
precoTotal	5	double		Preço dos produtos somado a entrega menos o desconto.
produtos	2	Produto		Produtos que o cliente adicionou ao carrinho.

Fonte: Elaborado pelo autor.

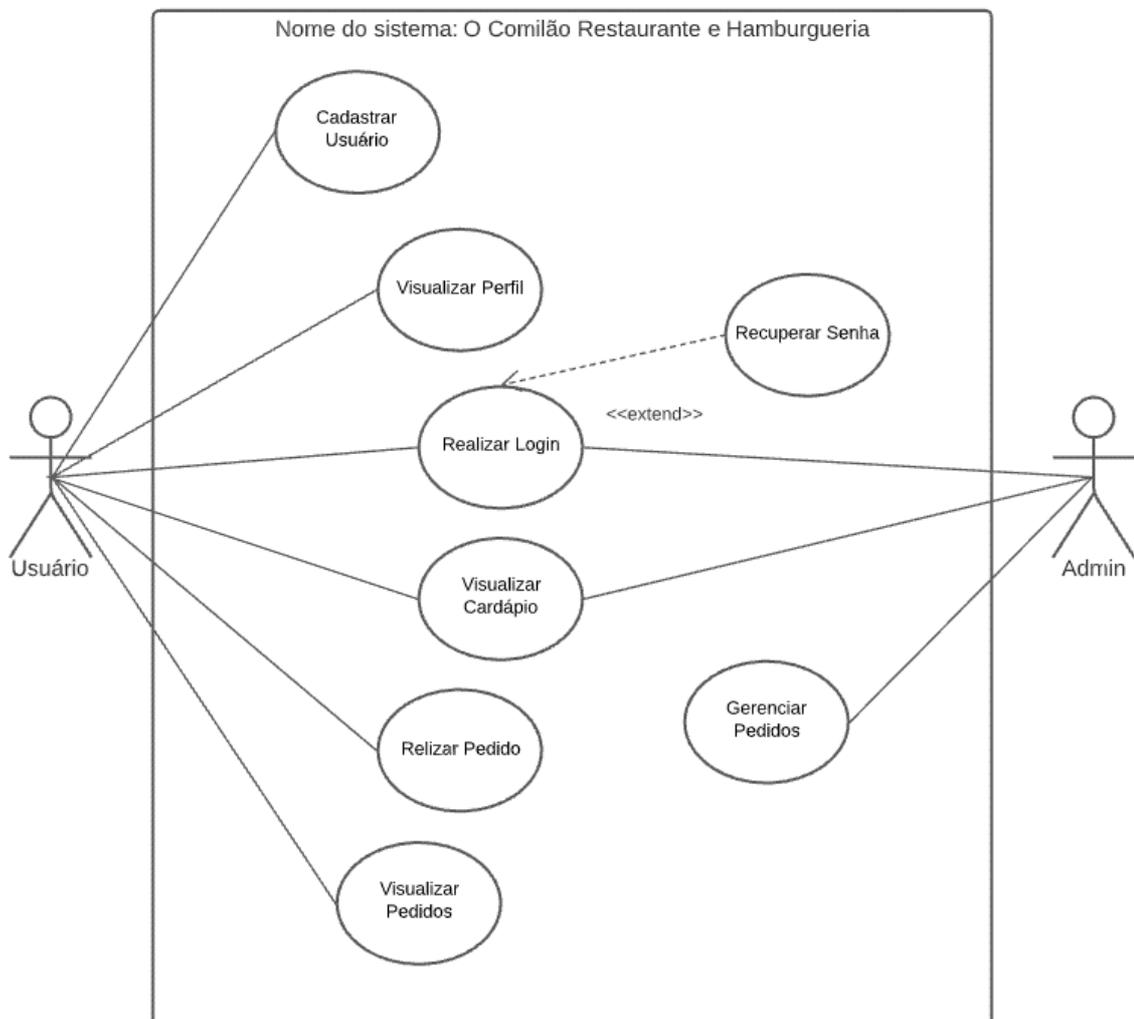
## 4 CASOS DE USO

Este capítulo apresenta o diagrama de casos de uso do sistema juntamente com os casos de uso descritivos.

### 4.1 Diagrama de casos de Uso

A Imagem 1 apresenta o diagrama de casos de uso do aplicativo.

Imagem 1 - Diagrama de casos de uso.



Fonte: Elaborado pelo autor.

## 4.2 Modelo Descritivo de casos de uso

Esta seção apresenta os casos de uso em modelo descritivo.

Quadro 17 - CSU 001 Visualizar cardápio.

**Identificador:** CSU 001

**Nome:** Visualizar cardápio.

**Requisito:** RF 004

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este Caso de Uso tem início quando o aplicativo é iniciado.

**Atores:** Usuário

**Pré-condições:** O ator deve ter acesso à internet.

**Pós-condições:** N.A.

**Cenário Principal:**

1. O aplicativo apresentará a tela Home.
2. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Usuário clica em algum produto.**

- O usuário seleciona algum produto do cardápio;
- O aplicativo mostra a tela do produto;
- Usuário clica no botão “Faça login para comprar”;
- O caso de uso CSU 002 é executado.

**1b – Fluxo alternativo – Usuário quer realizar login.**

- O usuário clica no botão de menu na parte superior esquerda da tela;
- O aplicativo abre a TabBar para usuário sem realizar login;
- Usuário clica em “Entre ou cadastre-se aqui”;
- O caso de uso CSU 002 é executado.

**1c – Fluxo alternativo – Usuário clica no carrinho.**

- O usuário clica no botão carrinho;
- O aplicativo mostra a mensagem “Faça login para acessar o carrinho”.

**Cenários de Exceção:**

- 2.1. E-mail preenchido não cadastrado no sistema;
  - 2.1.1. O sistema emite a mensagem de erro “Falha ao recuperar senha”.
- 2.2. Ator deixa o campo “E-mail” em branco;
- 2.3. O sistema emite a mensagem de erro “E-mail não preenchido”, abaixo do campo.

**Requisitos Relacionados:** RF 001

Fonte: Elaborado pelo autor.

## Quadro 18 - CSU 002 Realizar login.

**Identificador:** CSU 002

**Nome:** Realizar login.

**Requisito:** RF 002

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este caso de uso possibilita que os atores façam login no sistema.

**Atores:** Usuário

**Pré-condições:** O usuário deve ter cadastro no sistema e ter acesso a internet.

**Pós-condições:** Login do usuário deve ser efetuado.

**Cenário Principal:**

1. O sistema apresentará a tela de login;
2. O usuário preenche os campos “e-mail” e “senha”;
3. Ator seleciona a opção “Entrar”;
4. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Usuário quer realizar cadastro no sistema**

- O usuário seleciona a opção “Não tem uma conta?”;
- O caso de uso CSU 003 é executado.

**1b – Fluxo alternativo – Usuário quer recuperar sua senha**

- O usuário seleciona a opção “Esqueci minha senha”;
- O caso de uso CSU 004 é executado.

**1c – Fluxo alternativo – Usuário quer fazer login com Facebook**

- O usuário seleciona a opção “Entrar com Facebook”;
- Aplicativo abre um pop-up para o usuário fazer login no Facebook e aceitar os termos;
- Caso de uso se encerra.

**1d – Fluxo alternativo – Usuário quer fazer login com Gmail**

- O usuário seleciona a opção “Entrar com Google”;
- Aplicativo abre um pop-up para o usuário fazer login no Gmail e aceitar os termos;
- Caso de uso se encerra.

**Cenários de Exceção:**

- 3.1. O usuário deixa algum campo em branco;
  - 3.1.1. O sistema mostra uma mensagem de erro abaixo do campo que não foi preenchido.
- 3.2. O usuário digita um e-mail não cadastrado no sistema;
  - 3.2.1 O sistema emite a mensagem “E-mail ou senha incorretos”.
- 3.3. O usuário digita uma senha diferente da cadastrada no sistema;
  - 3.3.1 O sistema emite a mensagem “E-mail ou senha incorretos”.

**Requisitos Relacionados:** RF 001, RF 002, RF 003

Fonte: Elaborado pelo autor.

Quadro 19 - CSU 003 Cadastrar Usuário.

**Identificador:** CSU 003

**Nome:** Cadastrar usuário.

**Requisito:** RF001

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este caso de uso possibilita que o ator faça o cadastro no sistema.

**Atores:** Usuário

**Pré-condições:** O ator deve ter um e-mail válido e acesso à internet

**Pós-condições:** O sistema deve salvar o cadastro do usuário no Firebase.

**Cenário Principal:**

1. O sistema apresentará a Tela de Cadastro;
2. A tela exibirá um formulário de inscrição com campos para: nome, telefone, e-mail e senha;
3. Ator preenche o formulário e seleciona a opção “Registrar”;
4. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Usuário já está cadastrado**

- O usuário seleciona a opção “Já sou cadastrado!”;
- O caso de uso CSU 002 é executado.

**Cenários de Exceção:**

- 3.1a. O ator deixa algum campo em branco;
  - 3.1a.1. O sistema mostra uma mensagem de erro abaixo do campo que não foi preenchido.
- 3.1b. O ator colocar um número de telefone com o tamanho menor que 11;
  - 3.1b.1. O sistema mostra a mensagem “Número de telefone inválido, o telefone deve ter 11 dígitos”.
- 3.1c. O ator colocar um e-mail sem “@”;
  - 3.1c.1. O sistema mostra a mensagem “E-mail inválido.”

**Requisitos Relacionados:** RQ 001

Fonte: Elaborado pelo autor.

Quadro 20 - CSU 004 Recuperar senha.

**Identificador:** CSU 004

**Nome:** Recuperar senha.

**Requisito:** RF 003

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este Caso de Uso tem início quando o ator na Tela de Login, selecionar a opção “Esqueci minha senha”.

**Atores:** Usuário

**Pré-condições:** O ator deve ter cadastro no sistema e ter acesso à internet.

**Pós-condições:** O sistema deve enviar um e-mail para recuperação de senha.

**Cenário Principal:**

1. O sistema apresentará a Tela de recuperação de senha;
2. Ator preenche o campo “E-mail” e seleciona a opção “Recuperar Senha”;
3. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Usuário clica em cancelar**

- O usuário seleciona a opção “Cancelar”;
- O caso de uso CSU 002 é executado.

**Cenários de Exceção:**

- 2.1. E-mail preenchido não cadastrado no sistema;
  - 2.1.1. O sistema emite a mensagem de erro “Falha ao recuperar senha”.
- 2.2. Ator deixa o campo “E-mail” em branco;
- 2.3. O sistema emite a mensagem de erro “E-mail não preenchido”, abaixo do campo.

**Requisitos Relacionados:** RF 001

Fonte: Elaborado pelo autor.

Quadro 21 - CSU 005 Realizar pedido.

**Identificador:** CSU 005

**Nome:** Realizar pedido.

**Requisito:** RF 005

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este caso de uso possibilita que o ator faça um pedido no restaurante.

**Atores:** Usuário

**Pré-condições:** Usuário deve estar logado e ter acesso a internet.

**Pós-condições:** O sistema deve enviar o pedido para o banco de dados.

**Cenário Principal:**

1. Na tela Home, o usuário seleciona um produto;
2. O aplicativo apresenta a tela produto;
3. Usuário preenche o campo “Observação”, caso queira retirar algum ingrediente ou fazer alguma observação sobre o pedido;
4. Usuário aciona o botão “Adicionar ao carrinho”;
5. O aplicativo abre a tela carrinho;
6. O usuário seleciona o método de pagamento e digita o endereço de entrega;
7. O usuário aciona o botão finalizar pedido;
8. O aplicativo direciona o usuário para a tela pedidos;
9. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Usuário quer adicionar cupom de desconto**

- O usuário aciona o botão cupom de desconto;
- O aplicativo abre um campo para o usuário digitar o valor desejado.
- O caso de uso retorna ao passo 6.

**1b – Fluxo alternativo – Usuário quer solicitar troca**

- O usuário aciona o botão solicitar troca;
- O aplicativo abre um campo para o usuário digitar o valor desejado.
- O caso de uso retorna ao passo 6

**1c – Fluxo alternativo – Usuário quer aumentar a quantidade de itens adicionados**

- O usuário aciona o botão “+” ao lado do produto que deseja aumentar a quantidade;
- Aplicativo soma o valor da quantidade de produtos a mais adicionados ao valor final.

**1d – Fluxo alternativo – Usuário quer diminuir a quantidade de itens adicionados**

- O usuário aciona o botão “-” ao lado do produto que deseja diminuir a quantidade;
- Aplicativo subtrai o valor da quantidade de produtos a mais adicionados ao valor final.

**1e – Fluxo alternativo – Usuário quer remover um produto do carrinho**

- O usuário aciona o botão “Remover” ao lado do respectivo produto;
- Aplicativo remove o produto do carrinho.

**1f – Fluxo alternativo – Usuário quer adicionar outros produtos ao carrinho**

- O usuário aciona o botão voltar no topo da página;
- Aplicativo retorna para a página Home.

**Cenários de Exceção:**

- 6.1a. O ator deixa o campo endereço em branco;
- 6.1a.1. O sistema mostra uma mensagem de erro abaixo do campo endereço.

**Requisitos Relacionados:** RQ 003, RQ 004.

Fonte: Elaborado pelo autor.

Quadro 22 - CSU 006 Visualizar pedidos.

**Identificador:** CSU 006

**Nome:** Visualizar pedidos.

**Requisito:** RF 006

**Responsável:** Leonardo Moraes Rebelatto

**Descrição/Resumo:** Este caso de uso possibilita que o ator visualize os pedidos já realizados.

**Atores:** Usuário

**Pré-condições:** O ator deve estar logado e ter acesso a internet.

**Pós-condições:** N.A.

**Cenário Principal:**

<p>1. O sistema apresentará a tela pedidos; 2. Caso de uso se encerra.</p> <p><b>Cenários Alternativos:</b> N.A.</p> <p><b>Cenários de Exceção:</b> 1.1. Nenhum pedido realizado. 1.1.1. O sistema mostra a mensagem “Nenhum pedido realizado”.</p> <p><b>Requisitos Relacionados:</b> RF 005</p>
---

Fonte: Elaborado pelo autor.

Quadro 23 - CSU 007 Visualizar perfil.

<p><b>Identificador:</b> CSU 007 <b>Nome:</b> Visualizar perfil. <b>Requisito:</b> RF 001 <b>Responsável:</b> Leonardo Moraes Rebelatto <b>Descrição/Resumo:</b> Este caso de uso possibilita que o ator visualize o seu perfil de usuário. <b>Atores:</b> Usuário <b>Pré-condições:</b> O ator deve estar logado e ter acesso a internet. <b>Pós-condições:</b> N.A. <b>Cenário Principal:</b>  <ol style="list-style-type: none"> <li>1. O sistema apresentará a Tela de Home;</li> <li>2. O ator deve abrir acionar o botão do menu lateral;</li> <li>3. O sistema apresentará o menu lateral;</li> <li>4. O ator deve abrir acionar a opção perfil;</li> <li>5. O sistema apresentará a tela de perfil com as informações do usuário;</li> <li>6. Caso de uso se encerra.</li> </ol> <b>Cenários Alternativos:</b>  <b>1a – Fluxo alternativo – Usuário quer editar informações do perfil</b> <ul style="list-style-type: none"> <li>• O usuário edita suas informações;</li> <li>• O usuário aciona o botão “Editar Perfil”.</li> </ul> <b>Cenários de Exceção:</b> <b>Requisitos Relacionados:</b> RQ 001</p>
---

Fonte: Elaborado pelo autor.

Quadro 24 - CSU 008 Gerenciar pedidos.

<p><b>Identificador:</b> CSU 008 <b>Nome:</b> Gerenciar pedidos. <b>Requisito:</b> RF 008 <b>Responsável:</b> Leonardo Moraes Rebelatto <b>Descrição/Resumo:</b> Este caso de uso possibilita que o ator Admin gerencie os pedidos do restaurante. <b>Pré-condições:</b> O ator deve estar logado como administrador e ter acesso a internet.</p>
---

**Pós-condições:** Alterações devem ser atualizadas no banco de dados

**Cenário Principal:**

1. O sistema apresentará a Tela de Home;
2. O ator deve abrir acionar o botão do menu lateral;
3. O sistema apresentará o menu lateral;
4. O ator deve abrir acionar a opção Gerenciar Pedidos;
5. O sistema apresentará a tela de gerenciamento de pedidos com as informações de todos os pedidos no banco de dados;
6. Caso de uso se encerra.

**Cenários Alternativos:**

**1a – Fluxo alternativo – Ator quer mudar status do pedido**

- O ator aciona o botão avançar ou voltar status para alterar a fase que se encontra o pedido;
- O sistema troca o status do pedido no banco de dados;
- Caso de uso se encerra.

**1b – Fluxo alternativo – Ator quer remover pedido**

- O ator pode selecionar a opção “remover”;
- O sistema remove o pedido do banco de dados;
- Caso de uso se encerra.

**1b – Fluxo alternativo – Não há pedidos**

- O aplicativo apresentará a tela sem pedidos com a opção voltar;
- Caso de uso se encerra.

**Cenários de Exceção:**

**Requisitos Relacionados:** RQ 001

Fonte: Elaborado pelo autor.

## 5 DESENVOLVIMENTO DO APLICATIVO

Este capítulo apresenta as tecnologias usadas para o desenvolvimento do aplicativo, tal como a linguagem de programação Dart para as funcionalidades, o *Framework* Flutter para a criação das interfaces, e por fim o banco de dados Firebase para os serviços de autenticação de usuário e armazenamento de dados.

### 5.1 Dart

Dart é uma linguagem de programação criada pelo Google e tem o objetivo de oferecer a linguagem de programação mais otimizada para o desenvolvimento de aplicativos multiplataformas. Além de ser a base do Flutter, o Dart fornece a linguagem e os tempos de execução que impulsionam os aplicativos em Flutter (DART, 2020).

### 5.2 Flutter

Flutter é um *Framework* criado pelo Google que permite a criação de aplicativos para iOS, Android, Web e Desktop em um único código base. Além disso, traz facilidades como: (a) *Hot-Reload*, que funciona injetando arquivos de código fonte atualizados na *Dart Virtual Machine* em execução, mudando rapidamente as alterações para a visualização; e (b) desempenho nativo, que incorpora as diferenças críticas de plataforma, que mesmo sendo um código base para todas suas características, como rolagem, navegação, ícones e fontes, continuam diferentes (FLUTTER, 2020a).

#### 5.2.1 *Hot-Reload*

O *Hot-Reload*, quando acionado, faz com que a máquina *host* verifique o código editado desde a última compilação e recompila qualquer biblioteca que esteja com o código alterado. Os códigos atualizados são enviados para a máquina virtual (VM) Dart em execução. A VM então recarrega todas as bibliotecas atualizando as classes com as alterações e a estrutura do aplicativo é reconstruída, permitindo uma visualização rápida das mudanças (JOSEPHINE, 2019).

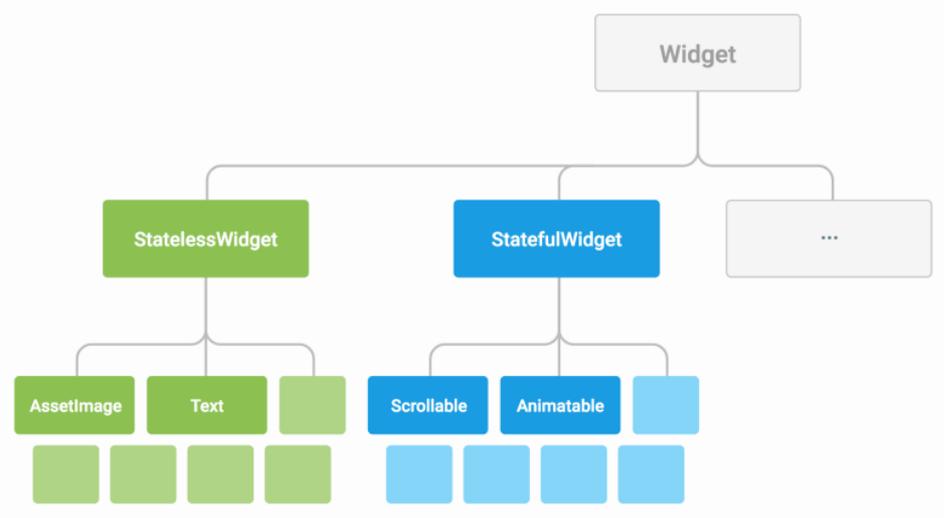
#### 5.2.2 *Widgets*

Os *widgets* são inspirados no React, com a ideia de que a interface seja construída a partir de widgets. Os widgets constroem a aparência do aplicativo de acordo com sua configuração e estados atuais. Ao criar o aplicativo, normalmente serão criados novos widgets

que serão subclasses de `StatelessWidget` ou `StatefulWidget`, dependendo se o widget gerencia seu estado (FLUTTER, 2020b).

O *Stateless Widget* é chamado somente uma vez, nenhuma alteração em qualquer variável pode chamá-lo novamente, para fazer alguma alteração ou redesenhar o widget uma nova instância do mesmo deve ser criada. Diferentemente, o *Stateful Widget* pode ser construído várias vezes, e cada vez que é chamado, dependendo de variáveis, funções ou eventos, pode ser desenhado de forma diferente (NUNES, 2019). A Imagem 2 mostra exemplos de widgets do tipo *Stateless* e *Stateful*.

Imagem 2 - Widgets.



Fonte: NUNES, 2019.

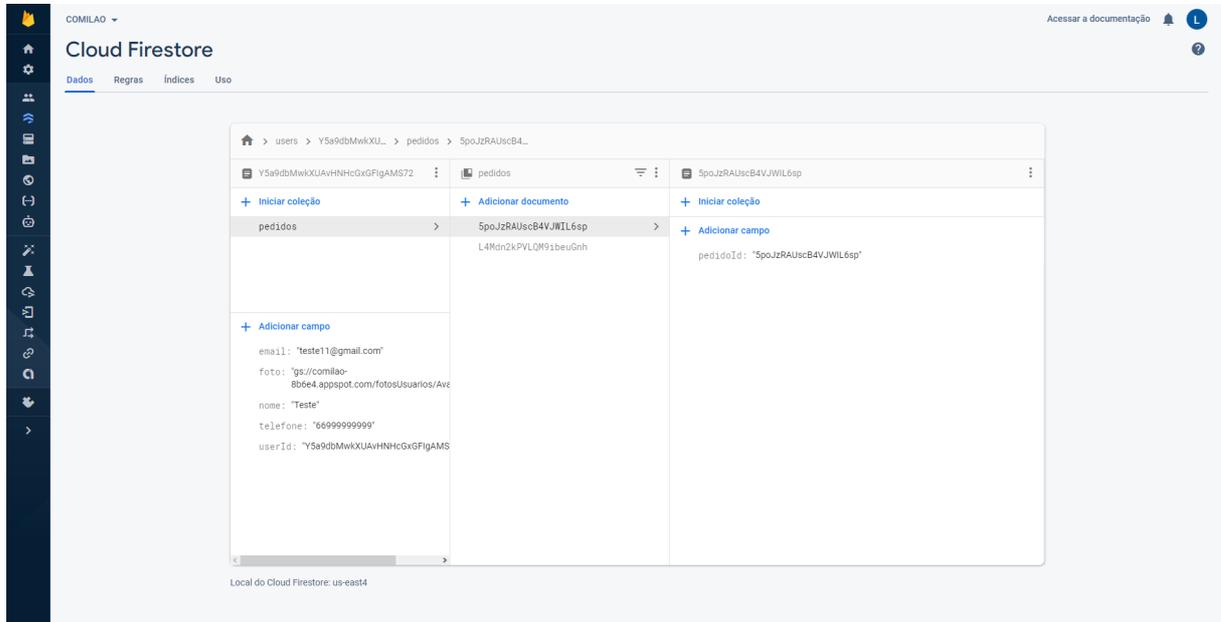
### 5.3 Firebase

O Firebase é uma plataforma desenvolvida pelo Google que fornece ferramentas necessárias para a criação de aplicativos, algumas delas foram usadas no desenvolvimento deste trabalho, sendo elas:

- **Firestore:** banco de dados em Nuvem NoSQL para armazenar e sincronizar dados por meio de *listeners* em tempo real, um exemplo da interface é apresentada na imagem 3 (FIREBASE, 2020a);
- **Firebase Authentication:** fornece serviços de *back-end*, SDKs de fácil uso e bibliotecas prontas para autenticar o usuário no aplicativo, visualização e gerenciamento de usuários como mostra a imagem 4. Autenticação de senhas, e-mails e telefones, serviço de recuperação de senhas automatizado com *templates* para fácil uso são mostrados na

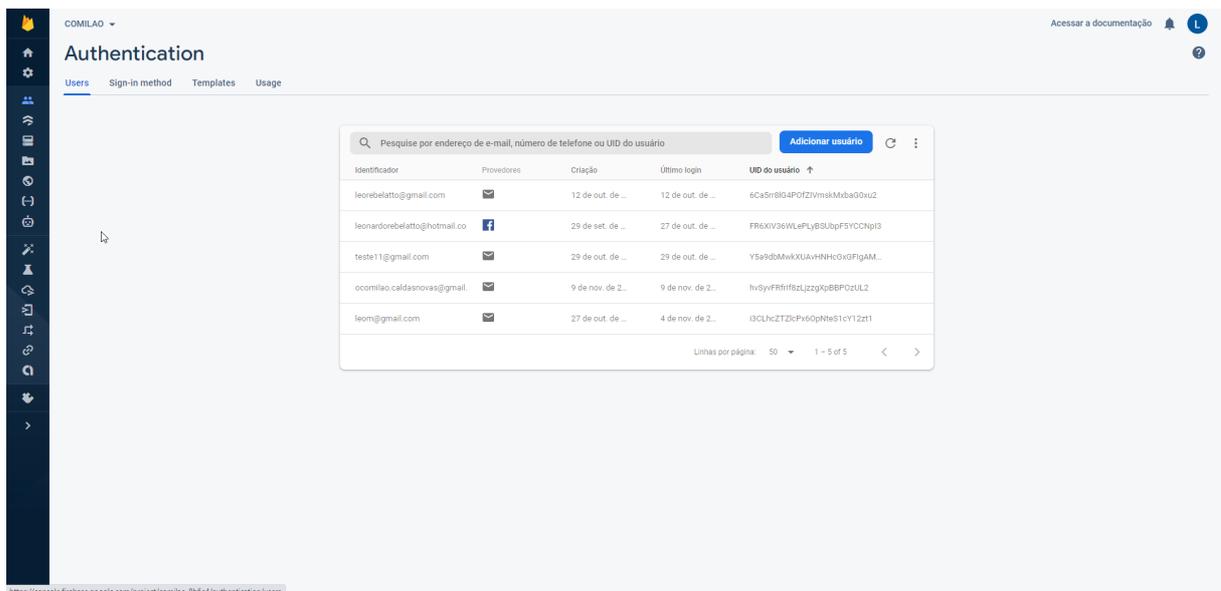
imagem 5, além de usar diferentes métodos de login como Facebook, Gmail, Twitter e outros como apresentados na imagem 6 (FIREBASE, 2020b).

Imagem 3 - Cloud Firestore.

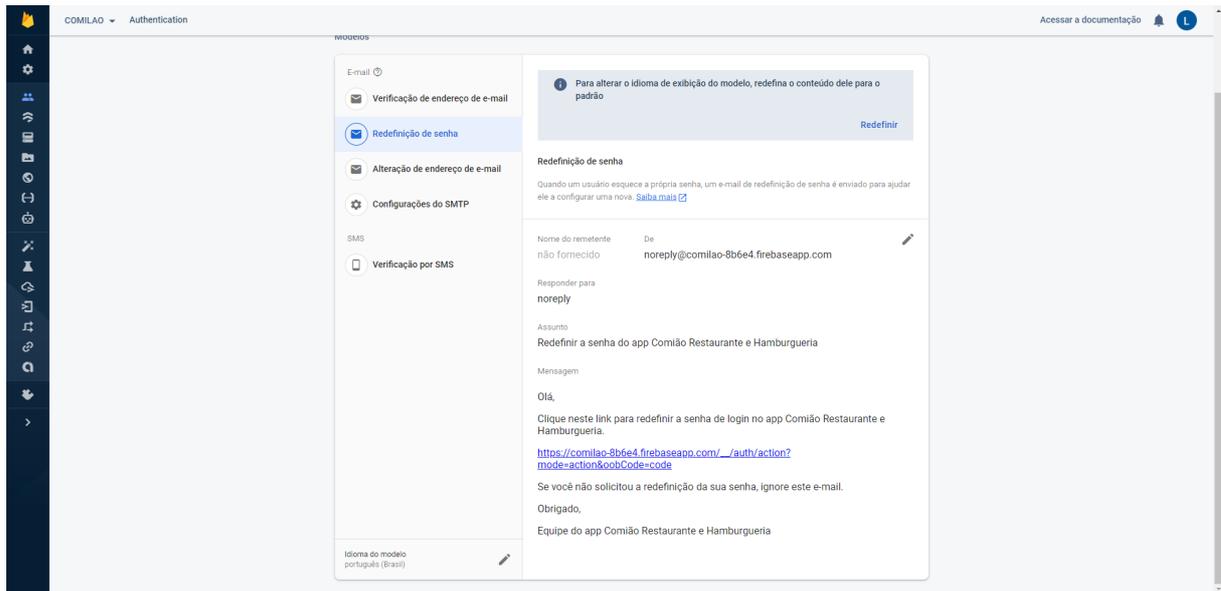


Fonte: Elaborado pelo autor.

Imagem 4 - Usuários do aplicativo.

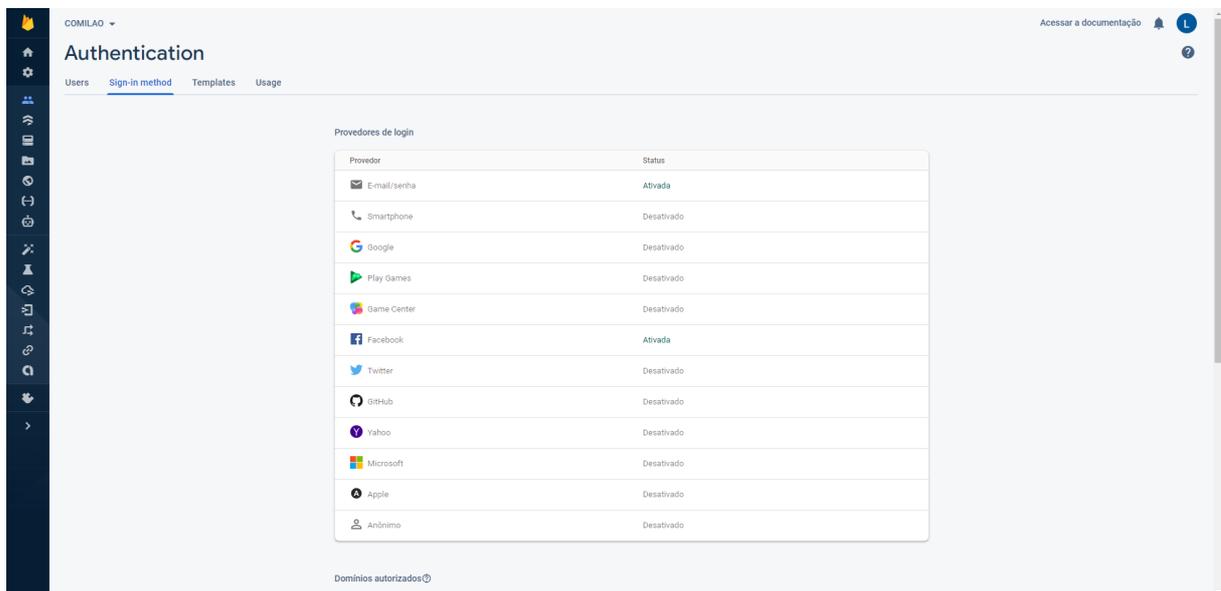


Fonte: Elaborado pelo autor.

Imagem 5 - *Template* redefinição de senha.

Fonte: Elaborado pelo autor.

Imagem 6 - Métodos de Login.

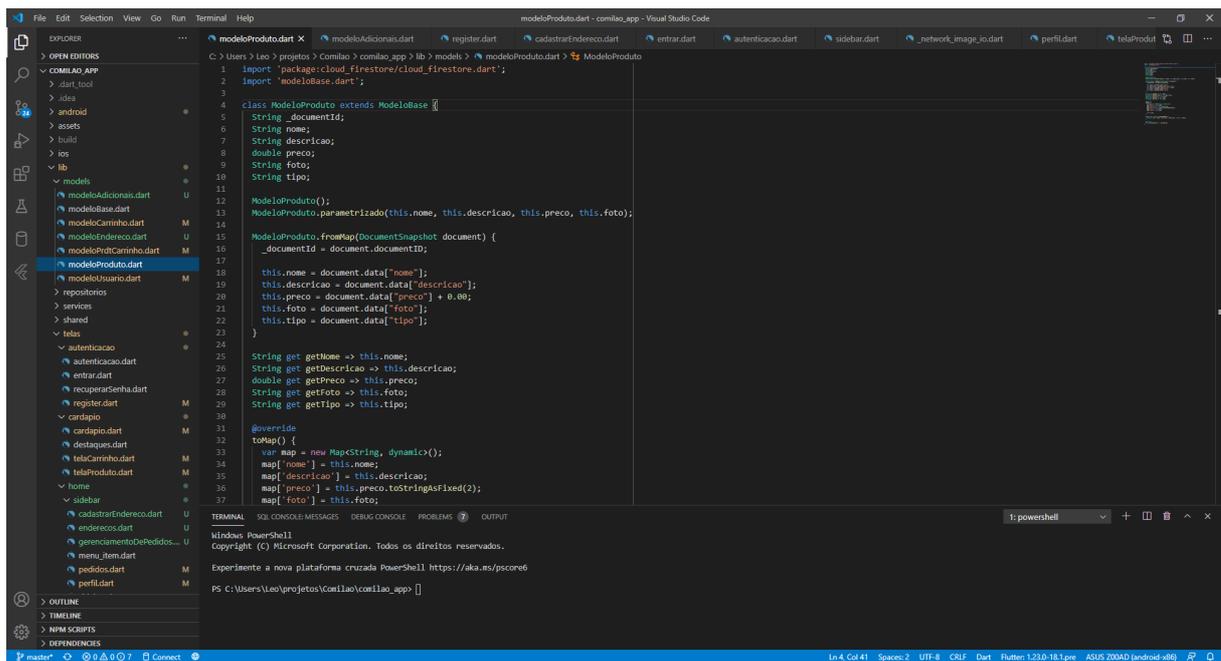


Fonte: Elaborado pelo autor.

## 5.4 Visual Studio Code

O Visual Studio Code, representado na Imagem 7, é um editor de código-fonte criado pela Microsoft e foi escolhido para desenvolver o aplicativo pelo conhecimento prévio do mesmo. Segundo Paim (2019), o Code é leve em termos de uso de memória quando comparado ao Android Studio.

Imagem 7 - Visual Studio Code.



Fonte: Elaborado pelo autor.

## 6 IMPLEMENTAÇÕES E RESULTADOS

Neste capítulo são apresentadas as telas do aplicativo, as descrições dos componentes das telas e as ações disponíveis ao usuário. O aplicativo tem três estados diferentes: (a) quando o usuário está sem efetuar login; (b) quando o usuário está logado como usuário cliente; e (c) quando o usuário está logado como administrador.

### 6.1 Telas Compartilhadas

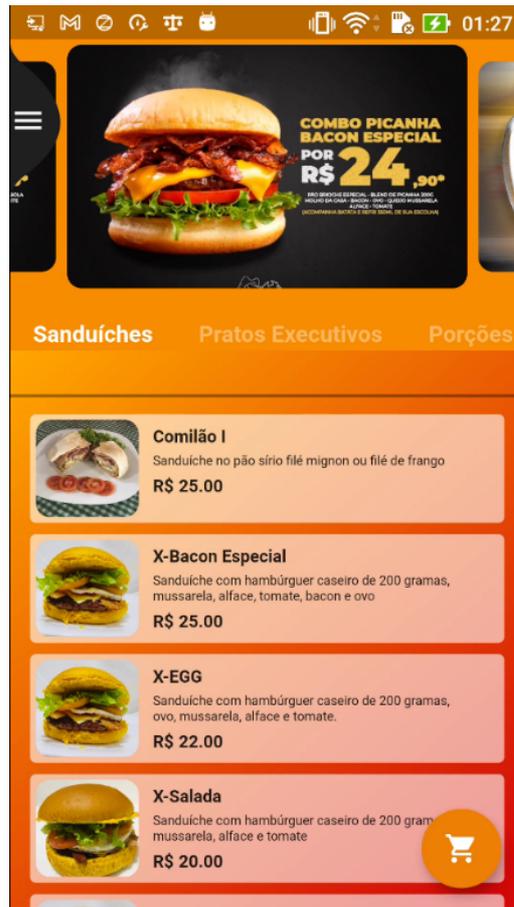
As telas compartilhadas aparecem nos três estados do aplicativo, cliente logado, admin logado ou usuário sem fazer login.

#### 6.1.1 Tela Home

A Imagem 8 mostra a tela inicial do aplicativo, esta tela possui dois botões, uma tabBar e uma lista de opções para selecionar os produtos:

- Botão carrinho: quando o usuário está logado, ao clicar nesse botão o usuário será redirecionado para a tela do carrinho. Quando o usuário não está logado, um *pop-up* informará que para visualizar o carrinho o usuário deverá realizar login no aplicativo;
- Barra de filtros: a barra terá a mesma quantidade de opções das categorias armazenadas no Firestore, se alguma categoria for adicionada, a barra aumentará a categoria dinamicamente. Quando o usuário clicar na opção da categoria desejada, a visualização da tela será redirecionada para o primeiro produto daquela categoria, de acordo com a Imagem 9;
- Botão superior lateral esquerdo: quando clicado, abrirá um menu lateral.

Imagem 8 - Tela Home.



Fonte: Elaborado pelo autor.

Imagem 9 - Tela Home posição pratos.



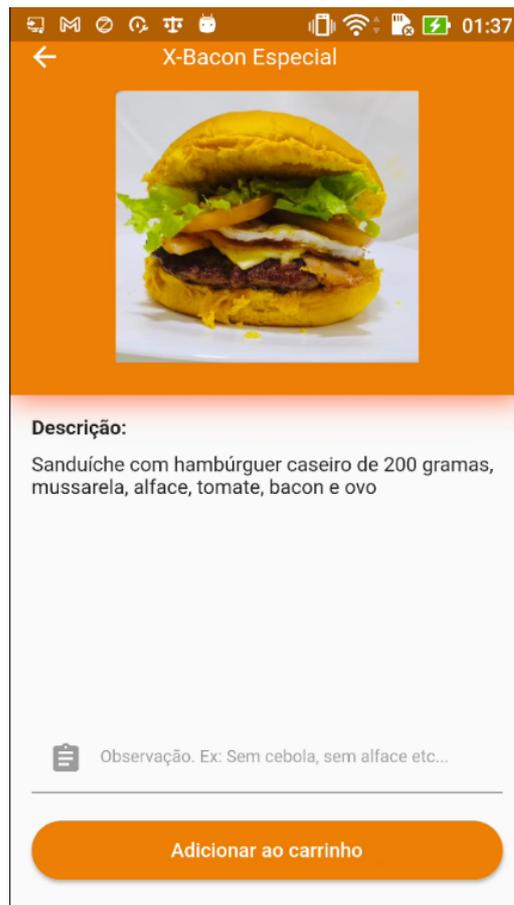
Fonte: Elaborado pelo autor

### 6.1.2 Tela produto

A tela do produto será apresentada quando o usuário clicar em algum produto na tela Home, contém um espaço para preenchimento e um botão, como mostra a Imagem 10:

- Campo Observação: deve ser preenchido caso o usuário deseje remover algum ingrediente do produto;
- Botão adicionar ao carrinho: ao clicar nesse botão o produto descrito na tela será adicionado ao carrinho e o usuário será redirecionado para a tela carrinho.

Imagem 10 - Tela produto.



Fonte: Elaborado pelo autor.

### 6.1.3 Tela de login

A tela de login possui dois campos para preenchimento e cinco botões, como mostra a Imagem 11:

- Campo “E-mail”: usuário deve informar o e-mail cadastrado;
- Campo “Senha”: usuário deve informar a senha com pelo menos 6 caracteres.
- Botão “Entrar”: se o e-mail e senha estiverem cadastrados, ao clicar nesse botão o usuário será redirecionado para a tela Home;
- Botão “Não tem uma conta?”: ao clicar nesse botão o usuário será direcionado para a tela de cadastro;
- Botão “Esqueci minha senha”: ao clicar nesse botão o usuário será direcionado para a tela de recuperação de senha;
- Botão “Entrar com Facebook”: ao clicar nesse botão o aplicativo abrirá um *pop-up* com a API do Facebook para o usuário fazer o login e aceitar os termos;

- Botão “Entrar com Google”: ao clicar nesse botão o aplicativo abrirá um *pop-up* com a API do Google para o usuário fazer o login e aceitar os termos.

Imagem 11 - Tela de login.



Fonte: Elaborado pelo autor.

#### 6.1.4 Tela de cadastro

A tela de cadastro possui quatro campos para preenchimento e dois botões, como mostra a Imagem 12:

- Campo “Nome”: usuário deve informar seu nome completo;
- Campo “Telefone”: usuário deve informar o número de telefone com 11 dígitos, sendo código de área e número;
- Campo “E-mail”: usuário deve informar o e-mail que deseja cadastrar, devendo conter “@”;
- Campo “Senha”: usuário deve informar a senha que deseja cadastrar com no mínimo 6 caracteres.

- Botão “Registrar”: se os dados forem validados, ao clicar nesse botão o usuário será redirecionado para a tela Home, logado com o usuário recém cadastrado;
- Botão “Já sou cadastrado!”: ao clicar nesse botão o usuário será direcionado para a tela de login.

Imagem 12 - Tela de cadastro.



21:24

O COMILÃO  
RESTAURANTE & HAMBURGUERIA

Nome

Telefone

E-mail

Senha

Registrar

Já sou cadastrado!

Fonte: Elaborado pelo autor.

### 6.1.5 Tela de recuperação de senha

A tela de recuperação de senha possui um campo para preenchimento e dois botões assim como mostra a Imagem 13:

- Campo “E-mail”: usuário deve informar o e-mail cadastrado;
- Botão “Recuperar Senha”: caso o e-mail informado esteja cadastrado, ao clicar nesse botão o aplicativo encaminhará um e-mail com um *link* para troca de senha;
- Botão “Cancelar”: ao clicar nesse botão o usuário será direcionado para a tela de login.

Imagem 13 - Tela de recuperação de senha.



Fonte: Elaborado pelo autor.

## 6.2 Telas com usuário sem efetuar login

Quando o usuário estiver sem efetuar login, ele terá acesso somente às telas compartilhadas e ao menu lateral.

### 6.2.1 Menu lateral com usuário sem efetuar login

Quando o usuário estiver sem efetuar login e abrir o menu lateral, terá acesso somente a um texto com a opção “Entre ou cadastre-se aqui”, para o usuário fazer login ou criar conta, de acordo com a imagem 14.

Imagem 14 - Menu lateral com usuário sem efetuar login.



Fonte: Elaborado pelo autor.

### 6.3 Telas com cliente logado

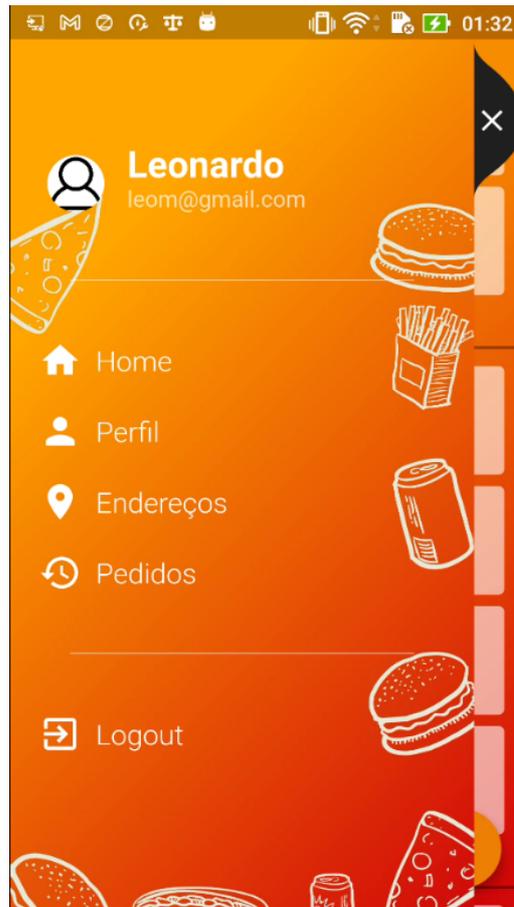
Quando o usuário estiver logado como cliente, ele terá acesso a todas as telas apresentadas nesta seção, além das telas compartilhadas.

#### 6.3.1 Menu lateral com usuário logado

Quando o usuário estiver logado e abrir o menu lateral ele terá acesso as suas informações nome e e-mail, e uma lista com cinco opções de acordo com a imagem 15:

- Opção 'Home': o menu será fechado e a tela Home será apresentada ao usuário;
- Opção 'Perfil': redireciona o usuário para a tela de perfil;
- Opção 'Pedidos': redireciona o usuário para a tela de pedidos;
- Opção 'Endereços': redireciona o usuário para a tela de endereços;
- Opção 'Logout': efetua *logout* do aplicativo e será apresentado ao usuário o menu lateral conforme a imagem 14.

Imagem 15 - Menu lateral com usuário logado.



Fonte: Elaborado pelo autor.

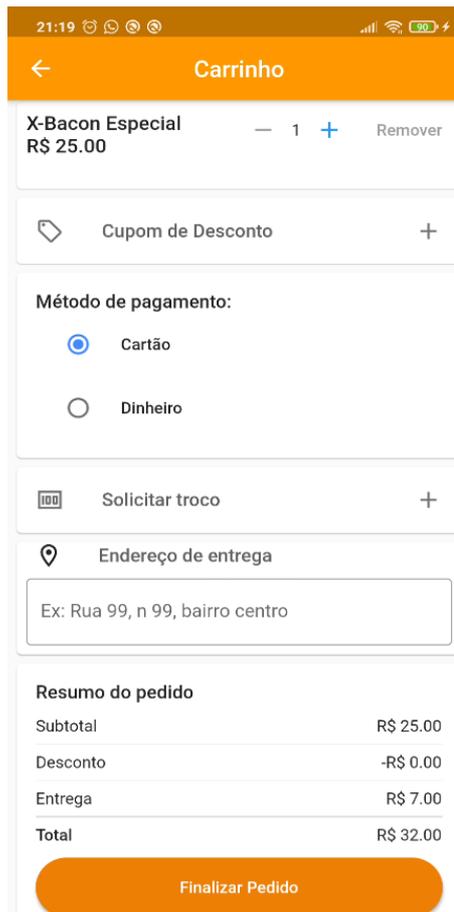
### 6.3.2 Tela carrinho

A tela de carrinho mostra todos os produtos que estão no carrinho, com uma variedade de botões e campos, como mostrado na imagem 16:

- Botões card Produto: ao clicar no botão “+” será adicionado mais uma unidade do produto à lista, caso o botão “-” for acionado o produto será reduzido em uma unidade e o botão “remover” remove o item do carrinho;
- Card Cupom de Desconto: ao clicar no “+” aparecerá um campo para preenchimento do cupom de desconto, assim que o código for digitado o sistema fará a validação e se for válido, a alteração no valor será realizada;
- Opção Método de pagamento: para finalizar o pedido o método de pagamento, cartão ou dinheiro, deve ser selecionado;
- Card Solicitar troco, ao clicar no “+” aparecerá um campo para preenchimento do troco desejado;

- Card Endereço de entrega, campo para preencher o endereço de entrega que o cliente deseja;
- Botão finalizar pedido: o sistema envia as informações do pedido para o banco de dados e redireciona o usuário para a tela pedidos.

Imagem 16 - Tela carrinho.

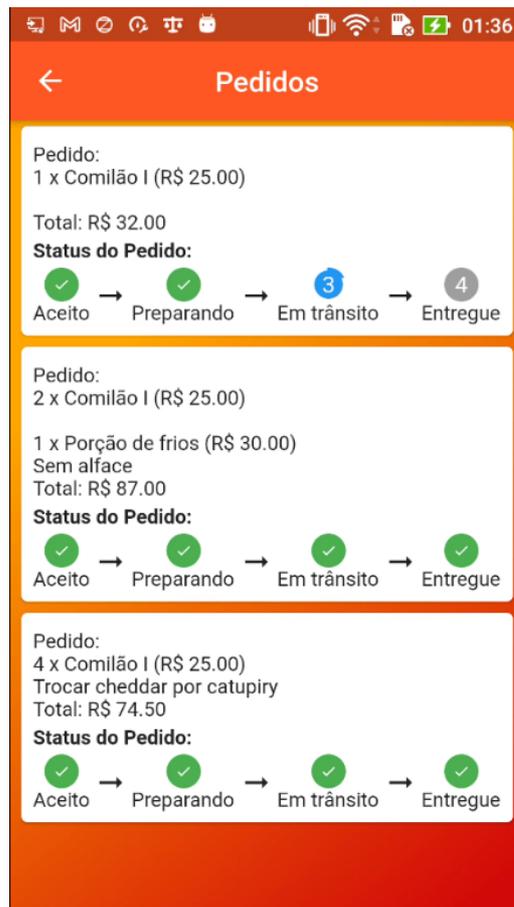


Fonte: Autoria própria.

### 6.3.3 Tela pedidos

A tela de pedidos é somente para visualização, o usuário pode acompanhar o andamento do seu pedido e visualizar pedidos anteriores, como mostra a Imagem 17.

Imagem 17 - Tela pedidos.



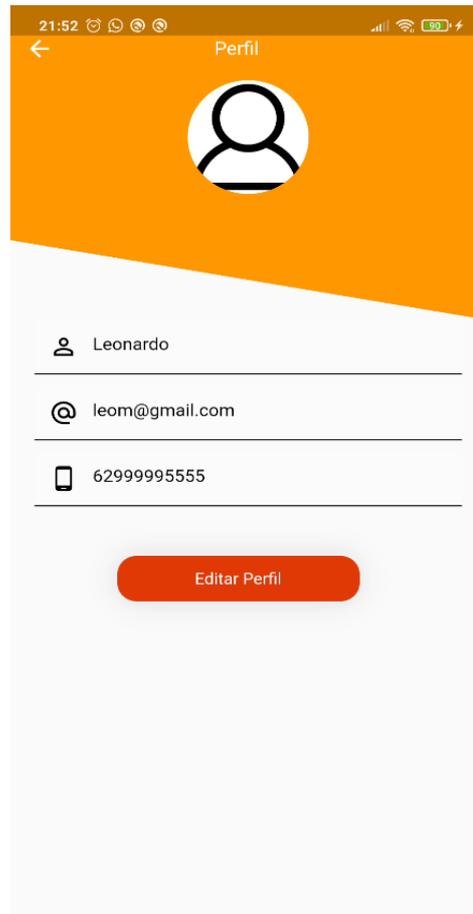
Fonte: Elaborado pelo autor.

### 6.3.4 Tela perfil

A tela de pedidos é para visualização e edição das informações de usuário, ela contém três campos para edição e um botão de acordo com a imagem 18.

- Campo nome contém o nome do usuário e pode ser editado;
- Campo e-mail contém o e-mail do usuário e pode ser editado;
- Campo telefone contém o telefone do usuário e pode ser editado;
- Gerenciar Pedidos: redireciona o usuário para a tela de gerenciamento de pedidos.

Imagem 18 - Tela Perfil



Fonte: Elaborado pelo autor.

## 6.4 Telas com usuário logado como administrador

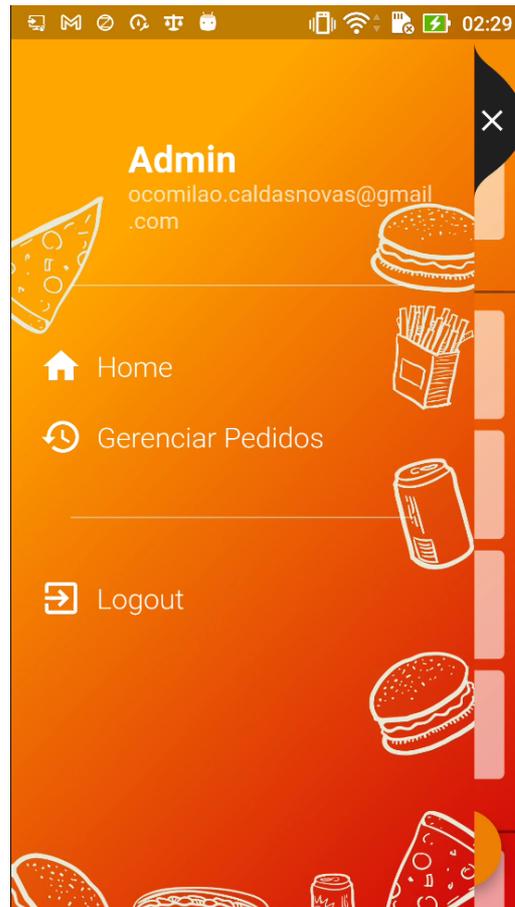
Quando o usuário estiver logado como administrador, ele terá acesso a todas as telas apresentadas nesta seção, além das telas compartilhadas.

### 6.4.1 Menu lateral com usuário logado como administrador

Quando o usuário estiver logado como administrador e abrir o menu lateral ele terá acesso as suas informações nome e e-mail, e uma lista com três opções de acordo com a imagem 19:

- Home: fecha o menu e apresenta a tela Home ao usuário;
- Gerenciar Pedidos: redireciona o usuário para a tela de gerenciamento de pedidos;
- Logout: efetua *logout* do aplicativo e o menu lateral apresentará a forma da imagem 14.

Imagem 19 - Menu lateral com usuário admin.



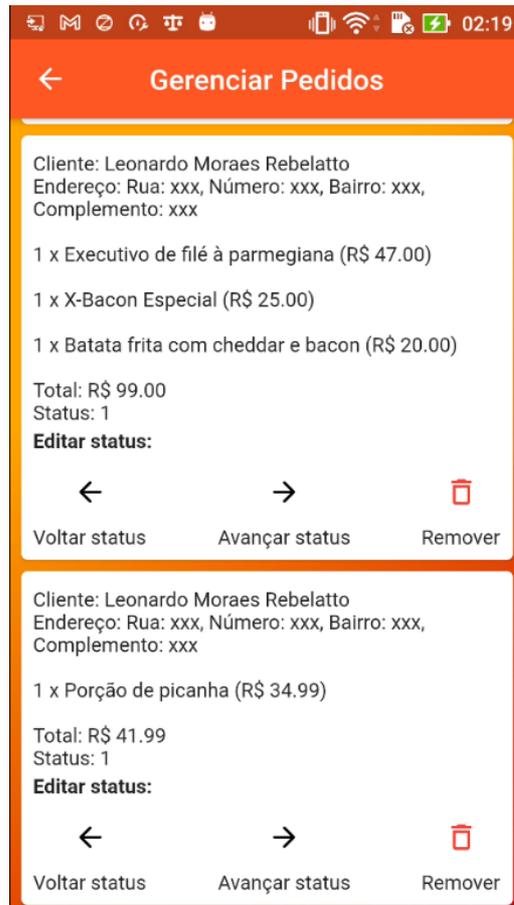
Fonte: Elaborado pelo autor.

#### 6.4.2 Tela Gerenciamento de pedidos

A tela de gerenciamento de pedidos mostra informações de todos os pedidos no restaurante e as opções para mudar seus status (1- Aceito, 2- Preparando, 3- Em transito, 4- entregue), como mostra a Imagem 20:

- Botão Voltar status: volta à fase anterior do acompanhamento do pedido;
- Botão Avançar status: avança à fase posterior do acompanhamento do pedido;
- Botão Remover: remove o pedido do banco de dados.

Imagem 20 - Tela gerenciamento de pedidos.



Fonte: Elaborado pelo autor.

## 7 CONSIDERAÇÕES FINAIS

Este trabalho teve como objetivo desenvolver e documentar um aplicativo para um restaurante, mostrando o cardápio virtual do estabelecimento e disponibilizando a opção para pedidos delivery. São mostradas as etapas e a documentação necessária para o desenvolvimento do aplicativo proposto, do levantamento de requisitos aos resultados da implementação, juntamente com as tecnologias usadas para a construção do aplicativo.

Os resultados obtidos satisfazem o objetivo do trabalho, visto que o aplicativo permite visualização do cardápio do restaurante e as informações de cada produto individualmente, além de permitir que os clientes façam os pedidos para entrega e o administrador do sistema consiga gerenciar estes pedidos pelo aplicativo.

Com o desenvolvimento do trabalho, algumas dificuldades foram encontradas, a principal delas foi a troca de tema por problemas com o uso de APIs que seriam utilizadas no tema anterior. Além do aprendizado de uma nova linguagem e a falta de conhecimento no desenvolvimento *mobile*.

### 7.1 Trabalhos Futuros

- Implementar opção de usar GPS para cadastrar endereço;
- Implementar comanda digital para os clientes que querem consumir no local;
- Implementar menu de adicionais para cada produto;
- Implementar pagamento online pelo aplicativo;
- Implementar função para gerenciar cardápio, quando logado como administrador;
- Implementar divisão da conta para pagamentos individualizados;
- Implementar acompanhamento de entrega por GPS;
- Implementar gerenciamento do aplicativo por meio de interface WEB;
- Implementar personalização do perfil.

## REFERÊNCIAS

BARRA, Daniela Couto Carvalho et al. **MÉTODOS PARA DESENVOLVIMENTO DE APLICATIVOS MÓVEIS EM SAÚDE: REVISÃO INTEGRATIVA DA LITERATURA.** 2017. Disponível em: <[http://www.scielo.br/scielo.php?script=sci\\_arttext&pid=S0104-07072017000400502&lng=en&nrm=iso](http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0104-07072017000400502&lng=en&nrm=iso)>. Acesso em 15 ago. 2020.

COSTA, Francisco José Carvalho; NERI, Marcos Antonio Lima. **PROPOSTA DE DESENVOLVIMENTO DE UMA APLICAÇÃO *MOBILE* PARA SERVIÇOS DE *DELIVERY* ALIMENTÍCIO E IMPACTOS DA COMPUTAÇÃO MÓVEL NO MERCADO GASTRONÔMICO.** 2019. Disponível em: <<https://sol.sbc.org.br/index.php/ercas/article/view/9045/8947>>. Acesso em 15 ago. 2020.

DART. **DART.** 2021. Disponível em: <https://dart.dev/overview>. Acesso em: 16 fev. 2021.

FIREBASE. **CLOUD FIRESTORE.** 2020a. Disponível em: <<https://firebase.google.com/docs/firestore>>. Acesso em: 10 nov. 2020.

FIREBASE. **FIREBASE AUTHENTICATION.** 2020b. Disponível em: <<https://firebase.google.com/docs/auth>>. Acesso em: 10 nov. 2020.

FLUTTER. **FLUTTER.** 2020a. Disponível em: <https://flutter.dev/>. Acesso em: 10 nov. 2020.

FLUTTER. **INTRODUCTION TO WIDGETS.** 2020b. Disponível em: <https://flutter.dev/docs/development/ui/widgets-intro>. Acesso em: 10 nov. 2020.

JOSEPHINE, Mareen. **UNDERSTANDING HOT-RELOAD IN FLUTTER.** 2019. Disponível em: <<https://medium.com/podiihq/understanding-hot-reload-in-flutter-2dc28b317036>>. Acesso em: 11 nov. 2020.

MENEZES, Gabrielli; YASSUDA, Saulo. **COMO O DELIVERY DE COMIDA MUDOU A ROTINA DA CIDADE.** 2020. Disponível em: <<https://vejasp.abril.com.br/comida-bebida/delivery-comida/>>. Acesso em 25 ago. 2020.

NEDEL, Matheus Berkenbrock. **SISTEMA INTEGRADO PARA ATENDIMENTO DE RESTAURANTES.** 2020. Disponível em: <[https://www.riuni.unisul.br/bitstream/handle/12345/10132/tcc\\_matheus.pdf?sequence=1&isAllowed=y](https://www.riuni.unisul.br/bitstream/handle/12345/10132/tcc_matheus.pdf?sequence=1&isAllowed=y)>. Acesso em 14 ago. 2020.

NUNES, Filipe. **E ESTE TAL DE FLUTTER, BORA INICIAR E ENTENDER.** 2019. Disponível em: <<https://medium.com/@FilipeFNunes/e-este-tal-de-flutter-bora-iniciar-e-entender-5278cbebe2ca>>. Acesso em 11 nov. 2020.

PAIM, Leonardo. **Utilizando o VS Code para criar Aplicativos Flutter**. 2019. Disponível em: <<https://medium.com/@leonardopaim/utilizando-o-vs-code-para-criar-aplicativos-flutter-windows-5074962ecfce>>. Acesso em: 11 nov. 2020.

## APÊNDICE A: BRIEFING DO SISTEMA

### 1. O que é o sistema?

É um aplicativo que fornece um cardápio digital e opção de pedidos *delivery* para o restaurante O Comilão de Caldas Novas – GO.

### 2. O que o sistema faz?

Ele permite que:

- Os clientes visualizem o cardápio pelo seu próprio *smartphone*.
- Os clientes que estão em casa façam o pedido por meio do aplicativo para entrega.

### 3. Como os usuários irão acessar o aplicativo?

Caso o usuário deseje apenas visualizar o cardápio, ele pode entrar anonimamente, porém, caso o usuário queira fazer seu pedido pelo aplicativo ele deve se cadastrar usando um e-mail e senha, Facebook ou pelo Gmail, e para concluir o cadastro deverá colocar um número de telefone válido.

### 4. Quem são os usuários?

Eles são os clientes do restaurante.

### 5. Como o cardápio deve ser apresentado?

O cardápio deve apresentar todos os produtos na mesma tela separados pelas suas categorias, com as promoções no topo da página, porém o usuário poderá pular a lista para a categoria do produto que deseja (Pizza, sanduíche, prato, bebida, etc.)

### 6. Quais os métodos de pagamento o sistema deve aceitar?

Pagamento na entrega em cartão ou em dinheiro selecionando o troco desejado.

## ANEXO A – TERMO DE AUTORIZAÇÃO DE PUBLICAÇÃO DE PRODUÇÃO ACADÊMICA



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE GOIÁS  
GABINETE DO REITOR

Av. Universitária, 1069 ● Sator Universitário  
Caixa Postal 86 ● CEP 74605-010  
Goiânia ● Goiás ● Brasil  
Fone: (62) 3946.1000  
www.pucgoias.edu.br ● reitoria@pucgoias.edu.br

### RESOLUÇÃO n° 038/2020 – CEPE

#### ANEXO I

#### APÊNDICE ao TCC

Termo de autorização de publicação de produção acadêmica

O(A) estudante Leonardo Moraes Rebelatto  
do Curso de Engenharia de Computação, matrícula 2015.1.0033.0352-7,  
telefone: 62 996633629 e-mail nado.rebelatto@gmail.com, na qualidade de titular dos  
direitos autorais, em consonância com a Lei nº 9.610/98 (Lei dos Direitos do autor),  
autoriza a Pontifícia Universidade Católica de Goiás (PUC Goiás) a disponibilizar o  
Trabalho de Conclusão de Curso intitulado  
Desenvolvimento de um aplicativo para gerenciamento de pedidos delivery de um  
restaurante, gratuitamente, sem ressarcimento dos direitos autorais, por 5  
(cinco) anos, conforme permissões do documento, em meio eletrônico, na rede mundial  
de computadores, no formato especificado (Texto (PDF); Imagem (GIF ou JPEG); Som  
(WAVE, MPEG, AIFF, SND); Vídeo (MPEG, MWV, AVI, QT); outros, específicos da  
área; para fins de leitura e/ou impressão pela internet, a título de divulgação da  
produção científica gerada nos cursos de graduação da PUC Goiás.

Goiânia, 17 de Fevereiro de 2021.

Assinatura do(s) autor(es): Leonardo Moraes Rebelatto

Nome completo do autor: Leonardo Moraes Rebelatto

Assinatura do professor-orientador: Ludmilla R. P. dos Santos

Nome completo do professor-orientador: Ludmilla Reis Pinheiro dos Santos