

---

**CFFOOD: Desenvolvimento de Aplicativo  
de Delivery para a Cidade de Córrego Fundo**

*Mateus Junior de Melo*

---



PROJETO DE CONCLUSÃO DE CURSO

Data de Depósito: 30/11/2021

Assinatura: \_\_\_\_\_

## **CFFOOD: Desenvolvimento de Aplicativo de Delivery para a Cidade de Córrego Fundo**

*Mateus Junior de Melo*

*Willyan Michel Ferreira*

Monografia apresentada ao Centro Universitário de Formiga UNIFOR/MG, como requisito parcial para obtenção do título de bacharel em Ciência da Computação, sob a orientação do Prof. Willyan Michel Ferreira

Unifor-MG - Formiga

11/2021



*À Deus, família e amigos*



# Agradecimentos

---

Palavras são ineficientes para expressar a alegria de um sonho realizado. Não existe sonho grande para DEUS, existe sim, minha participação, acreditando, e eis meu projeto concretizado. Um longo percurso repleto de desafios, conquistas, amadurecimento e aprendizado. Um tempo que me trouxe a oportunidade de vivenciar momentos únicos de grande satisfação pessoal.

Gratidão a DEUS, meu refúgio e fortaleza por guiar e abençoar sempre o meu caminho. Meu pai Ednei, que é meu exemplo de força, superação e persistência. A minha mãe Creusa, pelo exemplo de caráter e honestidade, pelo apoio, confiança, paciência e dedicação. Imprescindíveis, hoje e sempre! Obrigado por lutarem ao meu lado.

Aos meus amigos e amigas pelos momentos únicos que serão guardados. Aos mestres que acreditaram em mim e me destinaram uma contribuição inestimável ao meu aprendizado. A meu orientador Willyan Michel pela enorme força e por me direcionar aos caminhos certos em cada etapa deste trabalho. Enfim, agradeço e divido minha alegria com todos que de forma precisa alavancaram atos para o mérito desta conquista.

A todos e a cada um o MEU MUITO OBRIGADO!





*”Nunca se compare com ninguém neste mundo. Caso o faça, entenda que você estará insultando a si mesmo.”*

*Bill Gates*



# Sumário

---

<b>Lista de Figuras</b>	<b>ii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Considerações Iniciais . . . . .	1
1.2 Objetivos do Trabalho . . . . .	2
1.2.1 Objetivos Específicos . . . . .	2
1.3 Justificativa . . . . .	2
1.4 Estrutura da Monografia . . . . .	2
<b>2 Referencial Teórico</b>	<b>3</b>
2.1 Delivery . . . . .	3
2.1.1 Delivery em Córrego Fundo . . . . .	3
2.1.2 Delivery na Pandemia . . . . .	4
2.2 Aplicação Nativa . . . . .	4
2.3 Android Studio . . . . .	4
2.4 Java . . . . .	5
2.5 Firebase . . . . .	6
<b>3 Estudo da Arte</b>	<b>7</b>
3.1 Ifood . . . . .	7
3.2 PedidosAki . . . . .	8
3.3 Bom Delivery . . . . .	10
<b>4 Metodologia e Desenvolvimento</b>	<b>13</b>
4.1 Descrição do Problema . . . . .	13
4.2 Modelagem . . . . .	13
4.3 Requisitos Funcionais . . . . .	14
4.4 Construção do Banco de Dados . . . . .	18
4.5 Contrução do Aplicativo Móvel . . . . .	21

4.6	Tela de login de Empresas e Usuários . . . . .	22
4.7	Tela de Login com dados da Empresa . . . . .	23
4.8	Tela de Aviso . . . . .	24
4.9	Tela Inicial Empresa . . . . .	25
4.10	Menu Cadastrar Empresa . . . . .	26
4.11	Menu Adicionar Produto . . . . .	28
4.12	Relatório de Pedidos . . . . .	29
4.13	Menu Sair . . . . .	30
4.14	Tela Autenticação Usuário . . . . .	31
4.15	Tela de Aviso . . . . .	32
4.16	Tela inicial Usuário . . . . .	33
4.17	Cadastrar Dados do Usuário . . . . .	34
4.18	Menu Sair . . . . .	35
4.19	Efetuar Compra . . . . .	35
4.20	Tela Informar a Quantidade do produto . . . . .	37
4.21	Tela Mensagem Pagamento . . . . .	39
<b>5</b>	<b>Conclusão</b>	<b>41</b>
5.1	Trabalhos Futuros . . . . .	41

# Lista de Figuras

---

---

3.1	Telas do aplicativo Ifood - Fonte Google . . . . .	8
3.2	Empresas disponíveis no PedidoAki - Fonte Fortana Junior (2013) . . . . .	9
3.3	Central de Pedidos do PedidosAki - Fonte Fontana Junior (2013) . . . . .	9
3.4	Lista de Empresas do Bom Delivery - Fonte Caio Alcantara (2019) . . . . .	10
3.5	Lista de Produtos do Bom Delivery - Fonte Caio Alcantara (2019) . . . . .	11
4.1	Diagrama de Caso de Uso . . . . .	14
4.2	Autenticar Usuário . . . . .	14
4.3	Autenticar Empresa . . . . .	15
4.4	Cadastrar Usuário . . . . .	15
4.5	Cadastrar Empresa . . . . .	16
4.6	Adicionar Produtos . . . . .	16
4.7	Selecionar Produtos para a Compra . . . . .	17
4.8	Confirmar Pedido . . . . .	17
4.9	Ver Pedidos . . . . .	18
4.10	Firestore Authentication . . . . .	18
4.11	Tabelas . . . . .	19
4.12	Tabela Empresas . . . . .	19
4.13	Tabela Pedidos . . . . .	20
4.14	Tabela Empresas . . . . .	20
4.15	Tabela Produtos . . . . .	21
4.16	Tabela Usuários . . . . .	21
4.17	Tela Splash . . . . .	22
4.18	Tela Login . . . . .	23
4.19	Tela Login . . . . .	24
4.20	Tela Aviso . . . . .	25
4.21	Tela Inicial Empresa . . . . .	26
4.22	Tela Cadastro Empresa . . . . .	27

4.23	Tela Principal do Usuário . . . . .	28
4.24	Tela Cadastro Produto . . . . .	29
4.25	Tela Pedidos . . . . .	30
4.26	Tela Principal . . . . .	31
4.27	Tela Login com Dados do Usuário . . . . .	32
4.28	Tela Aviso . . . . .	33
4.29	Tela Inicial do Usuário . . . . .	34
4.30	Tela de Cadastro do Usuário . . . . .	35
4.31	Tela Principal do Usuário . . . . .	36
4.32	Tela do Usuário Produtos . . . . .	37
4.33	Tela Informar a Quantidade . . . . .	38
4.34	Tela Produtos . . . . .	39
4.35	Tela Informar Pagamento . . . . .	40

# Lista de Siglas

---

NUPENS - *Núcleo de Pesquisas Epidemiológicas em Nutrição e Saúde*

VRBIL - *Vale Refeição Benefícios ao Instituto Locomotiva*

CNDL - *Confederação Nacional de Dirigentes Lojistas*

DMO - *Digital Market Outlook*

RMA - *Rank My App*

ASP - *Active Server Pages*

IDE - *Integrated Development Environment*

APP - *Aplicativo*

GPS - *Global Positioning System*

IBGE - *Instituto Brasileiro de Geografia e Estatística*

JDK - *Java Development Kit*

JRE - *Java Runtime Environment*

SDK - *Software Development Kit*

JVM - *Java Virtual Machine*

JSE - *Java Standard Edition*

MBAAS - *Mobile BackEnd As A Service*

NoSQL - *Bancos de dados não relacionais*

JSON - *JavaScript Object Notation*

UI - *Interface de Usuário*

URL - *Uniform Resource Locator*





# Resumo

---

MELO, M. J. *CFFOOD: Desenvolvimento de aplicativo de Delivery para a cidade de Córrego Fundo*. Monografia (Graduação) — Centro Universitário de Formiga – Unifor-MG – Formiga, 2021.

Com a chegada da pandemia a entrega a domicílio passou de um diferencial para uma necessidade do mercado. Os aplicativos de *delivery* estão em alta, gerando empregos e contribuindo no crescimento do comércio. Porém, os principais aplicativos de *delivery* dificilmente oferecem serviço nas pequenas cidades do interior, onde as únicas opções de pedido com entrega são através de ligações telefônicas ou aplicativos de mensagem instantânea. Esse trabalho visa o desenvolvimento de um aplicativo de *delivery* com foco para cidades pequenas como Córrego Fundo, no intuito de facilitar a vida de quem precisa contratar algum tipo de serviço. O aplicativo foi desenvolvido em uma plataforma nativa, sendo disponibilizado para *Android*. O objetivo do aplicativo é auxiliar os comerciantes de cidades do interior a cadastrar e intermediar suas vendas de produtos através do aplicativo.

*Palavras-chave: Android, Cidades do Interior, Delivery e Pandemia.*



# Abstract

---

MATEUS, M. J. *CFFOOD: Delivery application development for the city of Córrego Fundo*. Monography (University Graduate) — Centro Universitário de Formiga – Unifor-MG – Formiga-MG, 2021.

With the arrival of the pandemic, home delivery changed from a differential to a market need. Delivery applications are on the rise, creating jobs and contributing to the growth of commerce. However, the main delivery applications hardly offer service in small towns in the interior, where the only options for ordering with delivery are via phone calls or via whatsapp. This work aims to develop a delivery application focused on small towns like Córrego Fundo, in order to make life easier for those who need to hire some type of service. The application was developed on a native platform, being made available for Android. The purpose of the application is to help merchants in inner cities to register and mediate their product sales through the application.

Keywords: Android, Inner Cities, Delivery e Pandemic.

---

# Introdução

---

## 1.1 Considerações Iniciais

A pandemia causada pelo coronavírus está causando vários impactos na sociedade e vem impondo drásticas mudanças na rotina da população mundial, várias medidas foram tomadas para conter a disseminação da covid-19 como o isolamento social, que cerca de 33,8 milhões de pessoas no Brasil estiveram rigorosamente isoladas. Com o isolamento alguns serviços ganham força como por exemplo o *delivery*, vem se destacando com um crescimento 60% de acordo com Instituto FoodService Brasil(CAMPOS, 2020) (IMPrensa, 2021).

Com isso a entrega a domicílio passou de um diferencial para uma necessidade do mercado em diferentes áreas com a chegada da pandemia no último ano, esse mercado se tornou ainda mais fundamental estimulando o maior consumo e influenciando hábitos da população, uma pesquisa nacional encomendada pela VR Benefícios ao Instituto Locomotiva(VRBIL) mostra que 81% dos estabelecimentos comerciais no Brasil passaram adotar o *delivery*, com isso milhares de estabelecimentos foram se adaptando e impondo como chave principal de renda (GUIMARAES, 2020).

Esse tipo de serviço já era utilizado em grandes centros, mais com a chegada da pandemia cidades pequenas e do interior também começaram a utilizar esse novo estilo de trabalho.

Córrego Fundo é uma cidade pequena do interior de Minas Gerais pouco mais de 6 mil habitantes, possui alguns serviços de entrega, porém tem uma grande dificuldade de organização dos pedidos, uma vez que os mesmos são feitos através de ligações telefônicas e via whatsapp.

Pesquisas feitas que em cidades abaixo de 80.000 habitantes os aplicativos de *delivery* são quase inexistentes ou não são incorporadas pelas lanchonetes e restaurante por causa das taxas (RIGOLO, 2019).

Tendo como objetivo contribuir para acesso em pequenos centros o presente trabalho irá propor a ideia de um aplicativo *mobile* chamado CFFOOD que auxiliará os habitantes em

fazerem seus pedidos via *delivery*.

## 1.2 Objetivos do Trabalho

O objetivo geral deste trabalho é desenvolver um aplicativo Android de *delivery* para a cidade de Córrego Fundo Minas Gerais.

### 1.2.1 Objetivos Específicos

A fim de atingir satisfatoriamente o objetivo , foram projetados alguns objetivos específicos como:

- Desenvolver uma plataforma intuitiva e de fácil gerenciamento.
- Cadastramento de Usuários e Empresas.
- Exibição de produtos.
- Compra através do aplicativo.

## 1.3 Justificativa

Pesquisa feita pelo Núcleo de Pesquisas Epidemiológicas em Nutrição e Saúde (NUPENS), milhares de brasileiros mudaram seus hábitos alimentares durante a pandemia e começaram a comprar por *delivery*, o mercado se tornou necessidade com a chegada da pandemia em 2020, método de entrega a domicílio passou de 49% antes da pandemia, para 81%, foi a categoria que mais cresceu no país pesquisa feita pela Confederação Nacional de Dirigentes Lojistas (CNDL)(JUNIOR, 2021) (FRANÇA, 2020).

Segundo o levantamento de pesquisa feito pela RankMyApp (RMA) houve um aumento de 30% no número de *downloads* de aplicativos por *delivery* em março de 2020 em comparação com o mesmo período do ano anterior, de acordo com o Digital Market Outlook (DMO), a previsão é de que o país deve atingir 39 milhões de usuários até 2024, os gastos com aplicativos de entregas, em especial para comidas, cresceram 149% durante a pandemia (FRANÇA, 2020) (DAU, 2021) (GORDON, 2021).

Tendo em vista o grande número de cidades do interior que não contém nenhum tipo de aplicativo de *delivery*. A presente monografia do aplicativo CFFOOD tem o objetivo de ajudar as cidades a terem um aplicativo onde possam pedir diretamente sua comida.

## 1.4 Estrutura da Monografia

Esta monografia está estruturada da seguinte maneira, o capítulo 1 descreveu a introdução com as considerações iniciais, objetivos do trabalho, objetivos específicos e a justificativa. O capítulo 2 apresenta o referencial teórico contendo os conceitos e tecnologias abordadas para este desenvolvimento. O capítulo 3 o estudo da arte que serviu de base para este projeto. O capítulo 4 mostra a metodologia utilizada e o desenvolvimento da aplicação. No capítulo 5 contém as conclusões, e os trabalhos futuros.

---

## Referencial Teórico

---

Esta sessão mostra as ferramentas e tecnologias que foram utilizadas para o desenvolvimento do aplicativo de delivery.

### 2.1 Delivery

*Delivery*, palavra de origem inglesa que passou a fazer parte do nosso vocabulário sobretudo por volta dos anos 2000, tem como seu principal significado trazer agilidade, comodidade e satisfação para os clientes (SOARES; LIMA, 2020.).

É uma estratégia de mercado, pois entrega os produtos diretamente ao cliente onde ele estiver, não sendo necessário sair de sua residência ou local de trabalho. (QUIRINO, 2016).

Atualmente o setor vem crescendo de forma significativa, sendo uma das formas de aumento no faturamento das empresas que estão investindo nesse método. A amplificação do acesso à *internet* e aplicativos são fatores apontados como contribuintes para a popularização da venda com *delivery*, pesquisa realizada pela Mobills (Startup de Gestão de Finanças Pessoais), mostra o crescimento 94,67%, das empresas na área de *delivery* se comparado ao período de janeiro a maio de 2019 (BULL, 2020).

#### 2.1.1 Delivery em Córrego Fundo

Córrego Fundo cidade com 6425 habitantes segundo Instituto Brasileiro de Geografia e Estatística (IBGE) situada no centro oeste de Minas Gerais, o comércio de alimentação na cidade é restrito com poucos estabelecimentos contendo lanchonetes restaurantes e padarias, os serviços de entrega são feitos por meio de ligações telefônicas e via whatsapp, por ser uma cidade pequena Córrego Fundo não tem nenhum tipo de aplicativo de *delivery*.(IBGE, 2021).

### 2.1.2 Delivery na Pandemia

A pandemia de covid-19 trouxe grandes mudanças para o setor alimentício, os estabelecimentos tiveram que se reinventar para conseguirem se manter e continuar seus negócios, empresas e estabelecimentos implantaram o método de *delivery* impulsionando suas vendas e aumentando sua área de atuação. Pesquisa nacional encomendada pela VR Benefícios (VRB) mostra que 81% dos estabelecimentos comerciais no Brasil passaram a fazer *Delivery* durante a pandemia e vão manter esta modalidade(GUIMARAES, 2020).

Apesar da crise, mais de 437 mil empresas de alimentação abriram no Brasil em 2020 durante a pandemia(GRANDI, 2021).

Essa necessidade de utilizar o *Delivery* demonstra um desenvolvimento de um público que não vive em apenas grandes cidades, mas também em cidades do interior, o que vem crescendo disparadamente devido à alta demanda, normalmente esses serviços são feitos através de aplicativos *mobile* de aplicação Nativa ou Híbrida (SILVA et al., 2021).

## 2.2 Aplicação Nativa

Aplicativos nativos são caracterizados por serem desenvolvidos em linguagem padrão da plataforma, em geral cada sistema operacional possui sua própria linguagem de programação, estes aplicativos não podem ser portados diretamente para outras plataformas pois será necessário uma nova implementação.(MATOS; BRITTO, 2016).

Utilizando desta linguagem os aplicativos nativos utilizam *frameworks* que fazem acesso direto a chamadas de sistema e recursos específicos do aparelho no qual se encontra em operação, tais como, sensor de proximidade, câmeras, GPS (Global Positioning System), contatos, *email* do usuário entre outros(EL-KASSAS, 2015).

Ao se acessar diretamente os *frameworks* do sistema operacional, algumas vantagens se tornam visíveis, podemos destacar a capacidade dos aplicativos de atingirem alta performance (menor consumo de bateria, memória ram, recursos de rede celular, espaço em disco) (MALAVOLTA, 2016).

Um aplicativo nativo pode ser desenvolvido para uma infinidade de usos, como para redes sociais, editor de imagens, jogos, setores corporativos, de prestação de serviços, entre muitos outros, são normalmente distribuídos para o público alvo através das respectivas lojas de aplicativos de cada plataforma (App Store, Play Store), podendo ser desenvolvido em vários tipos de IDE'S como o Android Studio onde contém todas as ferramentas necessárias para um desenvolvedor trabalhar. (TIAGO, 2021).

## 2.3 Android Studio

O Android Studio é um ambiente de desenvolvimento gratuito e multiplataforma, ele pode ser instalado em diferentes sistemas operacionais como windows, linux, macOS, possui uma interface atraente e vários recursos específicos para a plataforma Android que aumentam a produtividade na criação de seus aplicativos (ROBISON, 2015).

O Android Studio é o ambiente de desenvolvimento integrado *integrated development en-*

*vironment* (IDE), oficial para o desenvolvimento de aplicativos. Além do editor de código e das ferramentas de desenvolvedor avançadas do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na criação de apps Android como:

- Um sistema de compilação flexível baseado em *Gradle*
- Um emulador rápido com inúmeros recursos
- Um ambiente unificado que possibilita o desenvolvimento para todos os dispositivos *android*.
- A aplicação de mudanças para enviar mudanças de código e recursos ao aplicativo em execução sem reiniciar o aplicativo.
- Modelos de código e integração com *GitHub* para ajudar a criar recursos comuns de aplicativos e importar exemplos de código
- *Frameworks* e ferramentas de teste cheios de possibilidades
- Compatibilidade com C++ e NDK
- Compatibilidade integrada com o *google cloud platform* facilitando a integração do *Google Cloud Messaging* e do *App Engine*.

O Android Studio é uma IDE onde tem várias opções de linguagens para o desenvolvimento de aplicativos como o Java (DEVELOPERS, 2021).

## 2.4 Java

O Java foi lançado pela Sun Microsystems em 1995 sendo uma linguagem de programação e plataforma computacional que hoje pertence a *oracle*, tem como características a rapidez, segurança e confiabilidade estando em vários lugares como *notebooks*, celulares, vídeo *games*, sites e supercomputadores científicos. Java é capaz de escrever programas a serem executados em uma grande variedade de sistemas computacionais e dispositivos controlados por computador (AUGUSTO, 2021).

A linguagem Java por ser uma linguagem de alto nível e que foi desenvolvida para trabalhar em diferentes sistemas operacionais, ela é a base para praticamente todos os tipos de aplicações em rede e é o padrão global para o desenvolvimento e distribuição de aplicações móveis e incorporadas (ORACLE, 2021).

O Java tornou-se fundamental para os desenvolvedores, permitindo que eles escrevam um *software* em uma plataforma e o executem em qualquer outra plataforma por meio da máquina virtual Java (JVM), crie *softwares* robustos e eficientes para *smartphones*, *gateways*, microcontroladores, processadores remotos, sensores, módulos sem fio e praticamente qualquer outro aparelho eletrônico. A *oracle* tem dois produtos que implementam Java *Standard Edition* (JSE) que são Java SE *Development Kit* (JDK) e Java SE *Runtime Environment* (JRE). O JRE fornece as bibliotecas, a JVM, e outros módulos necessários para executar *softwares* escritos em Java, já o JDK contém tudo que está na JRE e mais algumas ferramentas, como os compiladores e depuradores necessários para o desenvolvimento de aplicativos Java. (ORACLE, 2021)



## 2.5 Firebase

O Firebase é um *MBaaS* (Mobile BackEnd As A Service) do google que fornece uma plataforma de ferramentas e serviços prontos para se integrar com o aplicativo. Com esse serviço não é necessário focar em regras de negócios, ou criar um serviço de autenticação, pois já é fornecido por esse provedor de *cloud*, possibilitando o rápido desenvolvimento da aplicação. Com foco em ser um *back-end* completo e de fácil usabilidade, essa ferramenta disponibiliza diversos serviços possibilitando a criação de aplicativos com mais agilidade, sem precisar gerenciar a infraestrutura, funcionalidades como análises, bancos de dados, mensagens e relatórios de erros(MORIBE, 2016).

Atualmente apresenta diversos recursos que facilitam o desenvolvimento, na qualidade e no crescimento das aplicações, dentre eles se destacam:

- **Realtime Database:** É um banco de dados NoSQL baseado em nuvem que sincroniza dados em todos os clientes em tempo real e oferece funcionalidade *off-line*, os dados são armazenados no banco de dados *realtime* como JSON e todos os clientes conectados compartilham uma instância, recebendo atualizações automaticamente com os dados mais recentes.
- **Authentication:** Autenticação de usuários de maneira simples, fornece serviços de *back-end* e bibliotecas de IU prontas para autenticar usuários em aplicativos. O serviço oferece suporte a autenticação usando senhas, números de telefone, provedores de identidade federados conhecidos, como *google*, *facebook* e *twitter*, entre outros.
- **Storage:** é um serviço de armazenamento de objetos avançado, simples e econômico criado para a escala do Google. Com os SDKs do *Firestore* para *Cloud Storage*, é possível usar a segurança do *Google* para fazer *upload* e *download* de arquivos nos apps do *Firestore*, independentemente da qualidade da rede (FIREBASE, 2021).

---

## Estudo da Arte

---

Nesta seção serão apresentados alguns trabalhos que de certa forma estão relacionados com os assuntos ou tecnologias deste projeto bem como as ferramentas de apoio e métodos usados pelos autores.

### 3.1 Ifood

O iFood é uma empresa brasileira fundada no ano de 2011, por 4 sócios, na cidade de São Paulo. É subsidiária de outras duas empresas, Just Eat (global multimilionária inglesa) e Movable (empresa de venture capital brasileira)(CANALTECH, 2011).

É um aplicativo que possui o cadastramento de estabelecimentos e usuários interessados pelo serviço de entrega de refeições. O iFood surgiu para modernizar o *delivery*, especializado em tele entrega, a ferramenta possui uma base de dados com o cardápio de vários estabelecimentos. É possível navegar entre os vários pratos oferecidos, visualizando informações de valor, ingredientes, tempo de preparo e avaliação do estabelecimento por outros clientes(GARBIN, 2014).

Após a escolha dos pratos é possível realizar um pedido para entrega, seguindo para uma área exclusiva de usuários cadastrados. Em seguida o pedido é enviado ao estabelecimento, junto com o endereço do cliente.

Na Figura 3.1 são apresentadas algumas telas do aplicativo, desde o início com a informação da localização, passando pela lista de estabelecimentos e chegando à tela de escolha do prato desejado.

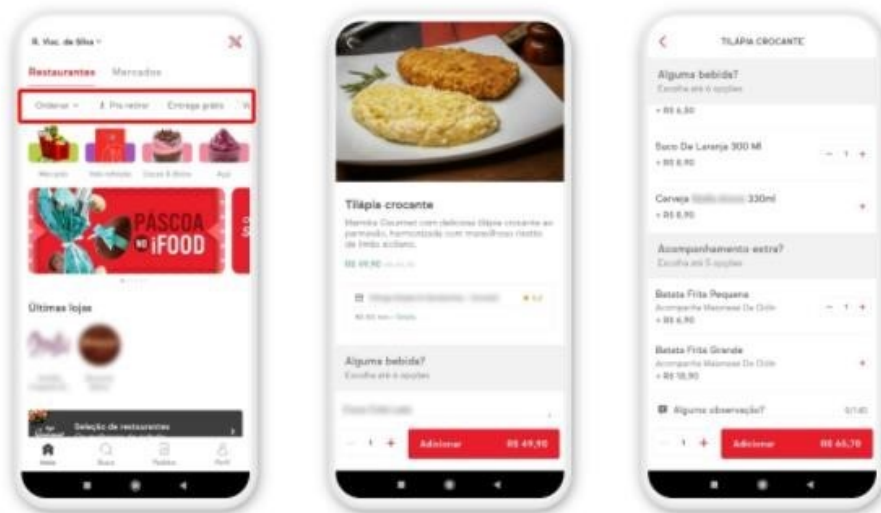


Figura 3.1: Telas do aplicativo Ifood - Fonte Google

Por se tratar de uma aplicação comercial não foi possível obter maiores detalhes sobre seu desenvolvimento. Para que o aplicativo possa ser executado em sistema Android a compatibilidade varia de acordo com os dispositivos. No iOS, ele é compatível com aparelhos que tenham versão 7.1 ou superior. No windows phone, a partir da versão 8.0 do sistema. Também é possível usar a versão web do iFood (TECHTUDO, 2016).

## 3.2 PedidosAki

Os estabelecimentos voltados a área alimentícia se mostram mais presentes. Apesar das várias novas empresas deste ramo, os clientes, por sua vez, nem sempre nem sempre conseguem opções variadas para solicitar o seu pedido como por exemplo *smartphone* ou computador, tendo que enfrentar linhas ocupadas ou trânsito. Os objetivos deste trabalho é criar um programa para *Android* usando Java onde o cliente pode solicitar os seus pedidos e um servidor Web usando *Active Server Pages (ASP) .NET*, *Model-ViewController (MVC)* e C para o gerenciamento da empresa fornecedora dos alimentos. (JUNIOR, 2015).

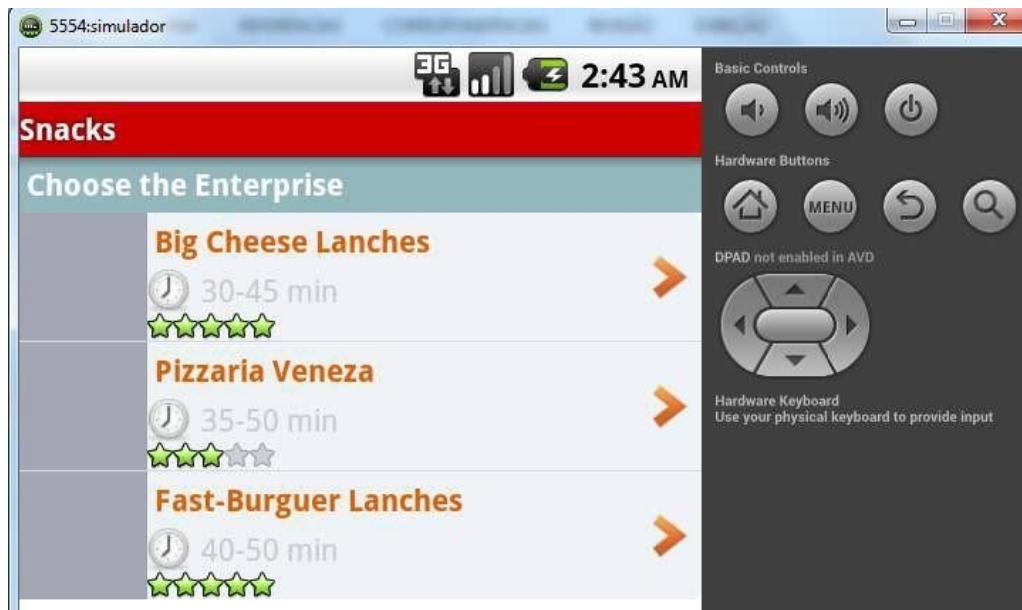


Figura 3.2: Empresas disponíveis no PedidoAki - Fonte Fortana Junior (2013)

A figura 3.2 exibe a apresentação das empresas disponíveis para a realização de pedidos próximas ao local em que o cliente está. Quando selecionada, exibe as opções de compras existentes no cardápio permitindo efetuar o pedido.

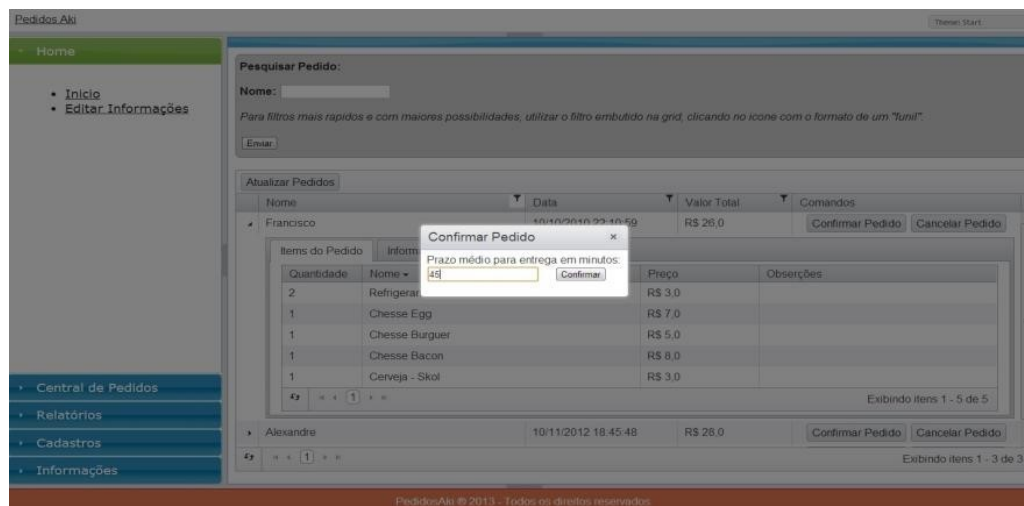


Figura 3.3: Central de Pedidos do PedidoAki - Fonte Fontana Junior (2013)

A figura 3.3 mostra a central de gerenciamento de pedidos que os estabelecimentos iram usar, nele é possível aceitar ou recusar algum pedido feito pelo cliente e informar o tempo em minutos para a entrega ser efetivada. Foram utilizados alguns componentes gráficos como *Telerik* e *jQuery UI* para aumentar a usabilidade do servidor. Não tem portabilidade para outros sistemas operacionais e quando está off-line fica totalmente fora de operação (JUNIOR, 2015).

### 3.3 Bom Delivery

O Bom Delivery é um aplicativo de *delivery*, com foco para cidades de pequeno porte, no intuito de auxiliar no acesso e venda nos estabelecimentos de alimentos e roupas, o aplicativo utiliza o mesmo princípio dos já existentes, porém, abrangendo qualquer cidade do Brasil, e intermediando a venda com entrega em domicílio de qualquer tipo de produto. O aplicativo traz vantagens tanto ao dono do estabelecimento que tem uma plataforma online para listagem de empresas mostrado na figura 3.4.(MELO et al., 2019).

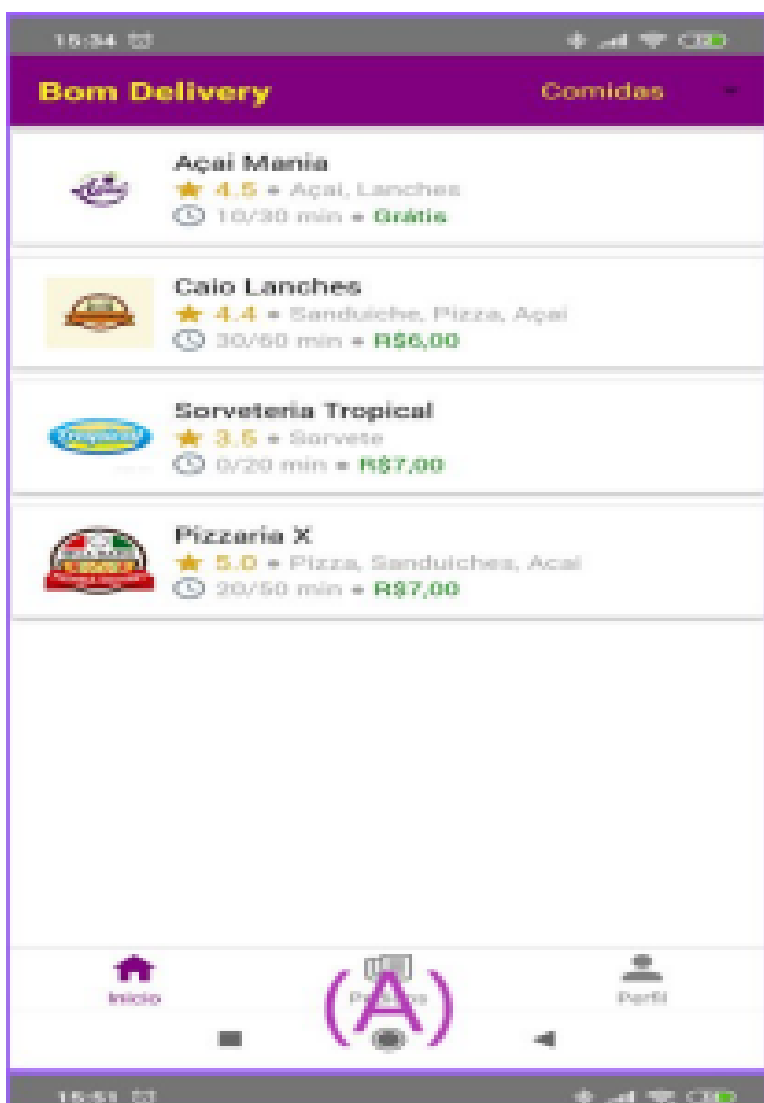


Figura 3.4: Lista de Empresas do Bom Delivery - Fonte Caio Alcantara (2019)

Quanto para o usuário que tem de forma simples o acesso a vários estabelecimentos e produtos através de seu smartphone mostrado na figura 3.5. O aplicativo foi desenvolvido em javascript, react e react nativa seu banco de dados foi utilizado o firebase. Além disso é multiplataforma sendo disponibilizado para em android e iOS, .(MELO et al., 2019).

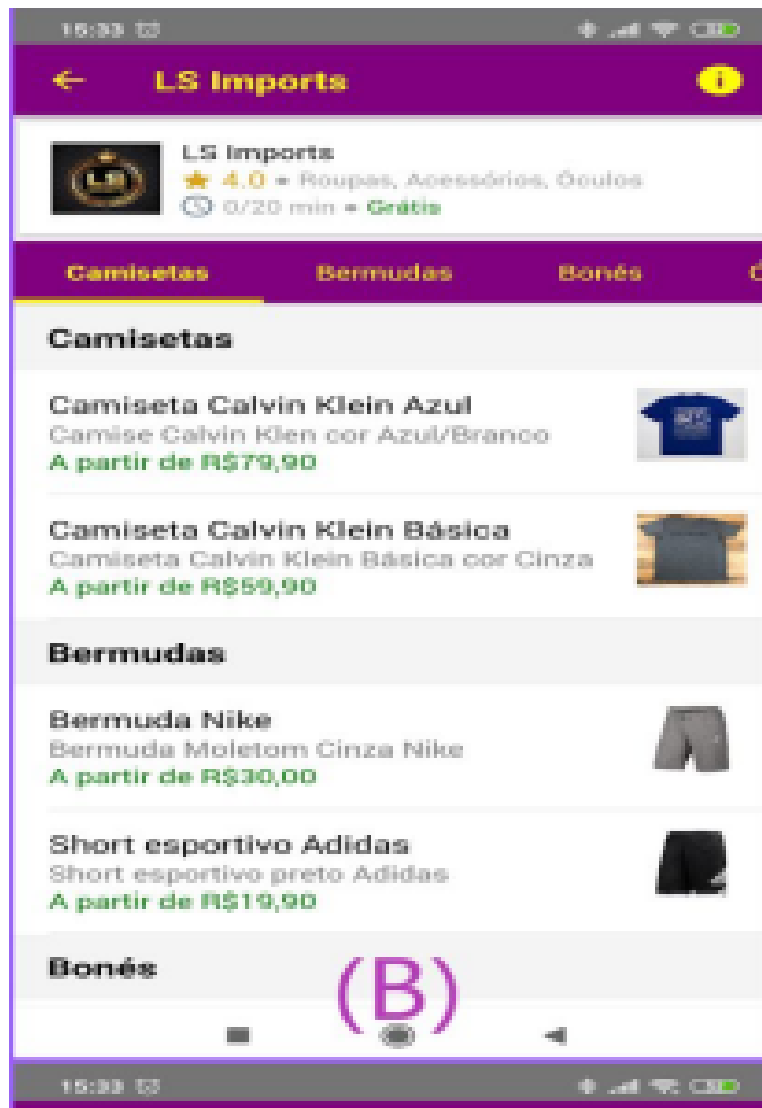


Figura 3.5: Lista de Produtos do Bom Delivery - Fonte Caio Alcantara (2019)



---

# Metodologia e Desenvolvimento

---

Este capítulo tem como o objetivo descrever o problema, a explanação de todo o processo de planejamento, levantamento dos requisitos, diagrama de caso de uso, construção do banco de dados e desenvolvimento realizado para a construção do aplicativo mostrando todas as telas passo a passo.

## 4.1 Descrição do Problema

Já existem vários aplicativos de sucesso como o iFood, UberEats e o Rappi, no ramo de *delivery*. Porém, não são todas as cidades do Brasil que utilizam a fundo essas tecnologias pois o custo é alto. As cidades de pequeno porte como por exemplo Córrego Fundo, onde existe um comércio restrito principalmente de comida, não tem nenhum tipo de aplicativo no ramo de *delivery*, as formas de venda dos produtos são apenas por ligações telefônicas e mensagens via *whatsapp*, deixando evidente a necessidade de um aplicativo de *delivery* na cidade.

## 4.2 Modelagem

Os diagramas de casos de uso fazem parte da UML e permitem a visualização do comportamento do sistema perante cada usuário, para que os desenvolvedores possam entender claramente suas operações e a implementação do sistema. Procura por meio de uma simples linguagem, possibilitar a compreensão o comportamento do sistema por qualquer pessoa, através de uma perspectiva do usuário. Ou seja, descreve o comportamento dos atores em relação as funcionalidades do sistema (FONSECA, 2015).

A Figura 4.1 mostra o diagrama de casos de uso do aplicativo CFFOOD.



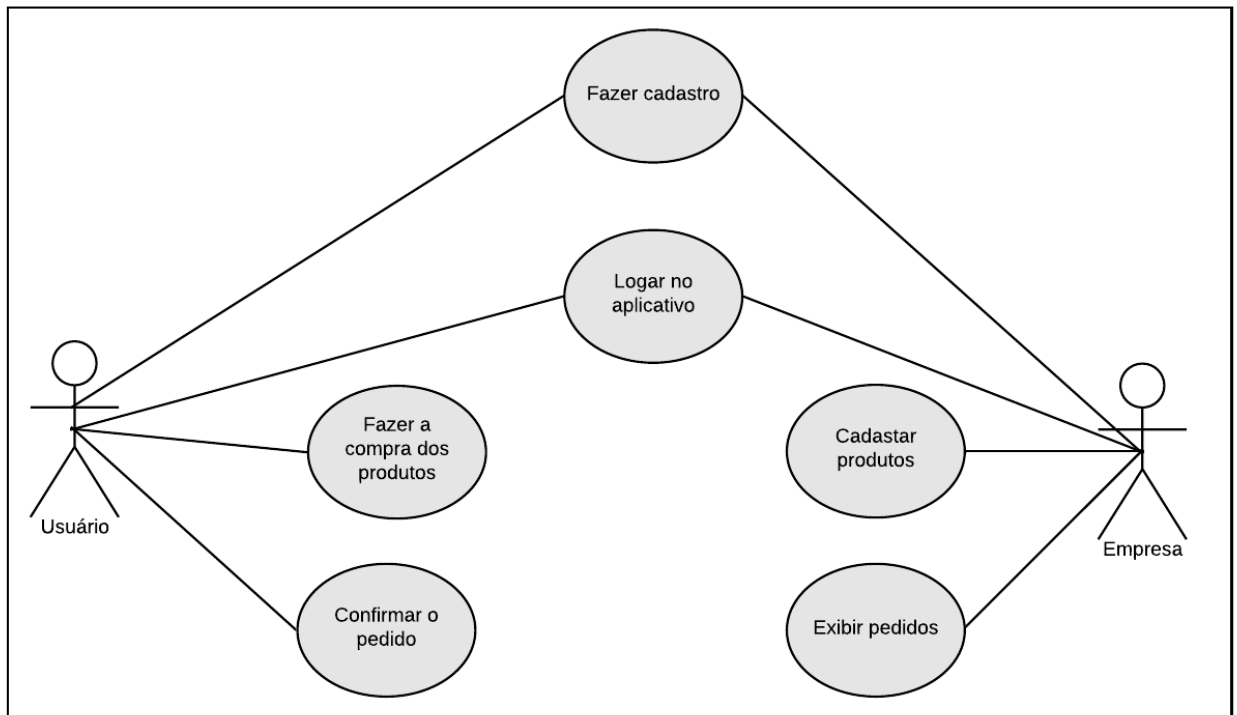


Figura 4.1: Diagrama de Caso de Uso

### 4.3 Requisitos Funcionais

Este caso de uso mostrado na figura 4.2 é executado pelo usuário. Como fluxo principal o usuário deve acessar a página de autenticação do aplicativo e inserir seu *e-mail* e senha e ativar primeiro *switch* no modo cadastrar, depois clicar no botão acessar para a realização da autenticação. Sem preenchimento dos dados não será possível entrar no aplicativo.

Descrição	Autenticação E-mail e Senha
Ator	Usuário
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Clicar e inserir e-mail e senha</li> <li>3) Ativar switch no modo cadastrar</li> <li>4) Clicar em Acessar</li> </ol>
Fluxo Alternativo	<ol style="list-style-type: none"> <li>1) Erro na aplicação</li> </ol>
Pré condições	Sem preenchimento dos dados impossível entra no aplicativo
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.2: Autenticar Usuário

Este caso de uso mostrado na figura 4.3 é executado pela empresa. Como fluxo principal a empresa deve acessar a tela de autenticação do aplicativo e inserir seu *e-mail* e senha e ativar

primeiro *switch* no modo cadastrar, ativar segundo *switch* no modo empresa, depois clicar no botão acessar para a realização da autenticação. Sem preenchimento dos dados não será possível entrar no aplicativo.

Descrição	Autenticação E-mail e Senha
Ator	Empresa
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Clicar inserir e-mail e senha</li> <li>3) Ativar switch no modo cadastrar</li> <li>4) Ativar switch no modo empresa</li> <li>5) Clicar em Acessar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Sem preenchimento dos dados impossível entra no aplicativo
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.3: Autenticar Empresa

Este caso de uso mostrado na figura 4.4 é executado pelo usuário. Como fluxo principal o usuário deve acessar a página inicial do aplicativo e clicar no botão cadastrar dados, onde serão preenchidos os dados do usuário como nome, endereço, Cep, cidade e telefone, depois clicar no botão salvar. Sem o preenchimento não será possível efetuar compras.

Descrição	Cadastrar dados do Usuário
Ator	Usuário
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Clicar cadastrar dados</li> <li>3) Inserir nome, endereço, Cep, cidade e telefone</li> <li>4) Clicar em Salvar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Impossível efetuar compras sem o cadastramento dos dados
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.4: Cadastrar Usuário

Este caso de uso mostrado na figura 4.5 é executado pela empresa. Como fluxo principal a empresa deve acessar a página inicial do aplicativo e clicar no botão cadastrar empresa, onde serão preenchidos os dados da empresa como logo, nome, tempo de entrega, taxa, endereço

e telefone, depois clicar no botão salvar. Sem o preenchimento não será possível cadastrar produtos.

Descrição	Cadastrar dados da Empresa
Ator	Empresa
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a Aplicação</li> <li>2) Clicar Cadastrar Empresa</li> <li>3) Inserir imagem, nome, tempo, taxa, endereço e telefone</li> <li>4) Clicar em salvar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Impossível adicionar produtos sem a autenticação dos dados da empresa
Pós condições	Saber se o android é compatível com a aplicação

Figura 4.5: Cadastrar Empresa

Este caso de uso mostrado na figura 4.6 é executado pela empresa. Como fluxo principal adicionar produtos onde deve acessar a página inicial do aplicativo e clicar no botão add produtos, onde serão preenchidos os dados dos produtos como nome, descrição e preço, depois clicar no botão salvar. Sem o preenchimento dos dados não será possível adicionar produtos.

Descrição	Adicionar Produtos
Ator	Empresa
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Clicar Add Produtos</li> <li>3) Informar o nome, descrição e preço</li> <li>4) Clicar em Salvar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Sem os dados impossível cadastrar os produtos
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.6: Adicionar Produtos

Este caso de uso mostrado na figura 4.7 é executado pelo usuário. Como fluxo principal selecionar produtos para a compra, onde deve acessar a página inicial selecionar a empresa, assim será exibido todos seus produtos depois selecionar o produto desejado informar a quantidade que deseja comprar e confirmar. Sem adicionar a quantidade de itens não será possível fazer a compra.

Descrição	Selecionar produtos para a compra
Ator	Usuário
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Selecionar a empresa</li> <li>3) Selecionar o produto</li> <li>4) Informar a quantidade desejada</li> <li>5) Confirmar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Sem adicionar a quantidade de itens impossível fazer a compra.
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.7: Selecionar Produtos para a Compra

Este caso de uso mostrado na figura 4.8 é executado pelo usuário. Como fluxo principal a confirmação do pedido, onde deve acessar a página compra e clicar no botão confirmar pedido onde aparecerá uma caixa de texto onde o usuário irá informar o valor exato que irá pagar pelo pedido.

Descrição	Confirmar o pedido
Ator	Usuário
Fluxo Principal	<ol style="list-style-type: none"> <li>1) Acessar a aplicação</li> <li>2) Clicar Confirmar Pedido</li> <li>3) Informar a forma de pagamento</li> <li>4) Confirmar</li> </ol>
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Informa a forma de pagamento
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.8: Confirmar Pedido

Este caso de uso mostrado na figura 4.9 é executado pela empresa. Como fluxo principal é a exibição dos pedidos, onde se deve acessar a página inicial e clicar no botão pedidos. Se não houver nenhum pedido feito pelo usuário não haverá nenhum pedido exibido.

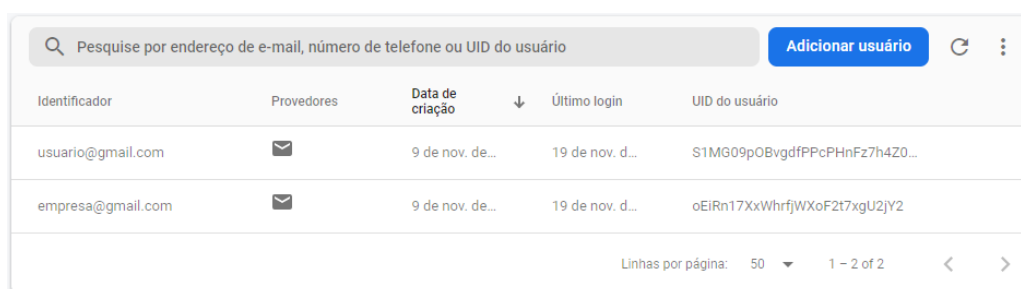
Descrição	Ver os pedidos
Ator	Empresa
Fluxo Principal	1) Acessar a aplicação 2) Clicar no botão pedidos
Fluxo Alternativo	1) Erro na aplicação
Pré condições	Haver pedidos feitos pelos usuários
Pós condições	Saber se o Android é compatível com a aplicação

Figura 4.9: Ver Pedidos

Após a descrição do problema, levantamento e análise dos requisitos e do diagrama de caso de uso, o processo de criação do banco de dados foi iniciado.

## 4.4 Construção do Banco de Dados

Para a construção do banco de dados do aplicativo foi utilizado o Firebase que consistem em um banco de dados não-relacional localizado na nuvem onde apresenta diversas funções que facilitam o desenvolvimento de uma aplicação como a *Authentication* onde serão armazenados todos os *e-mail* e senhas, *Realtime Database* será responsável pela construção de todas as tabelas e suas informações e a *Storage* onde será armazenado todas as imagens.



Identificador	Provedores	Data de criação	Último login	UID do usuário
usuario@gmail.com	✉	9 de nov. de...	19 de nov. d...	S1MG09pOBvgdfPPcPHnFz7h4Z0...
empresa@gmail.com	✉	9 de nov. de...	19 de nov. d...	oEiRn17XxWhrfjWXoF2t7xgU2jY2

Figura 4.10: Firebase Authentication

Usuários e empresas devem efetuar *login com e-mail* e senha para autenticar no aplicativo, com os dados preenchidos, a função *authentication* ficará responsável pelo armazenamento dos *emails* e senhas, informando data de criação, o ultimo *login* e criando um id para cada empresa usuário cadastrados mostrado na figura 4.10.

A função *Realtime Database* mostrada na figura 4.11 será responsável por todas as tabelas do aplicativo armazenando as informações das empresas, usuários, produtos e pedidos.

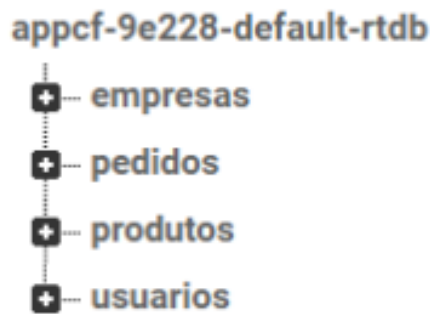


Figura 4.11: Tabelas

A Figura 4.12, tabela empresas ficará responsável pelo armazenamento dos dados das empresas como contato, endereço, nome, precoEntrega, tempo e a url da imagem.



Figura 4.12: Tabela Empresas

Na figura abaixo 4.13 e a 4.14, a tabela pedidos, ficará responsável pelo armazenamento dos dados dos pedidos contendo os campos de nome, Cep, cidade, endereço, contato, método de pagar e a obs. Contendo outra tabela dentro chamada itens com os campos nome, preço e quantidade.

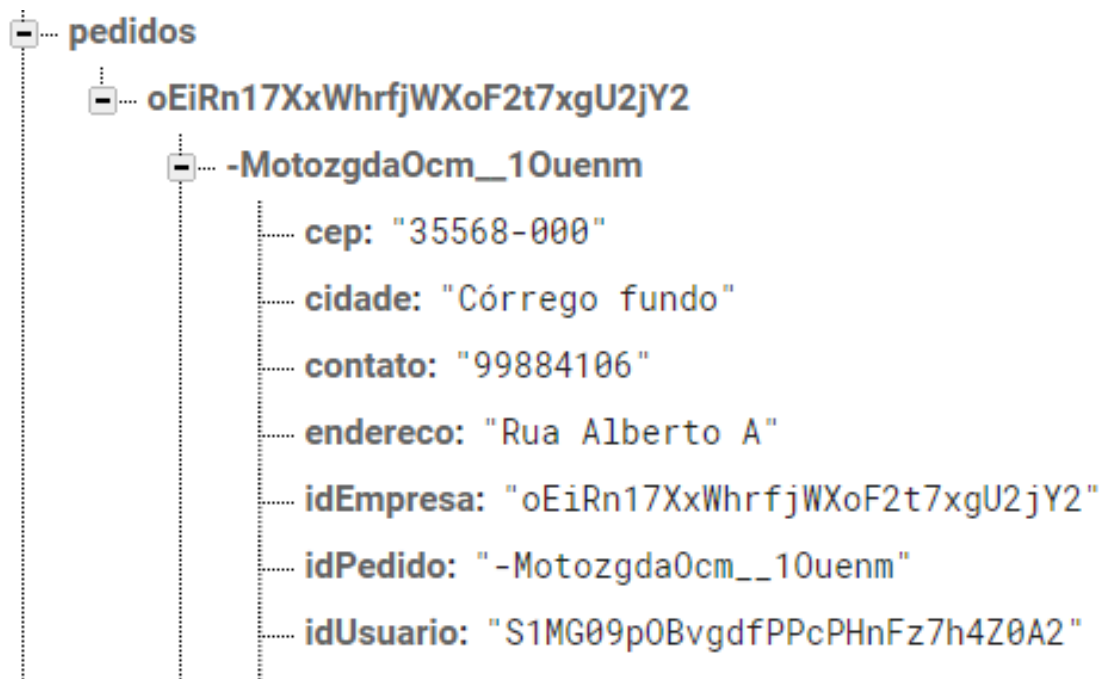


Figura 4.13: Tabela Pedidos

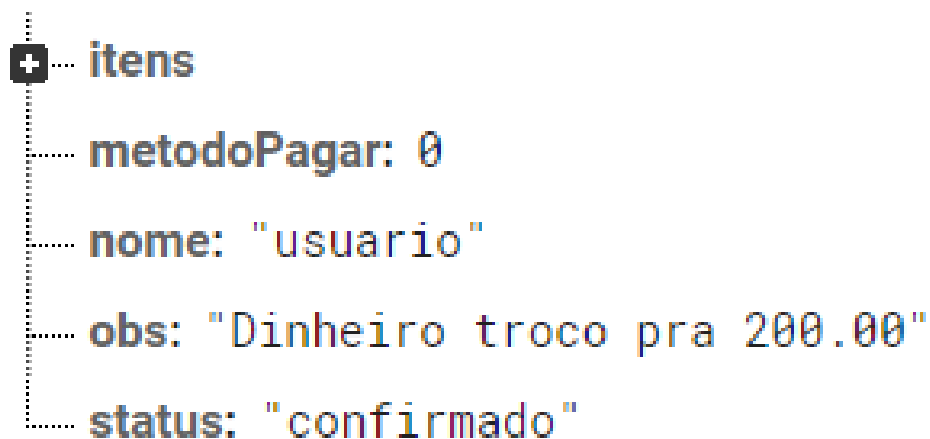


Figura 4.14: Tabela Empresas

A tabela produto ficará responsável pelo armazenamento das informações dos produtos como nome, descrição e preço mostrado na figura 4.15.

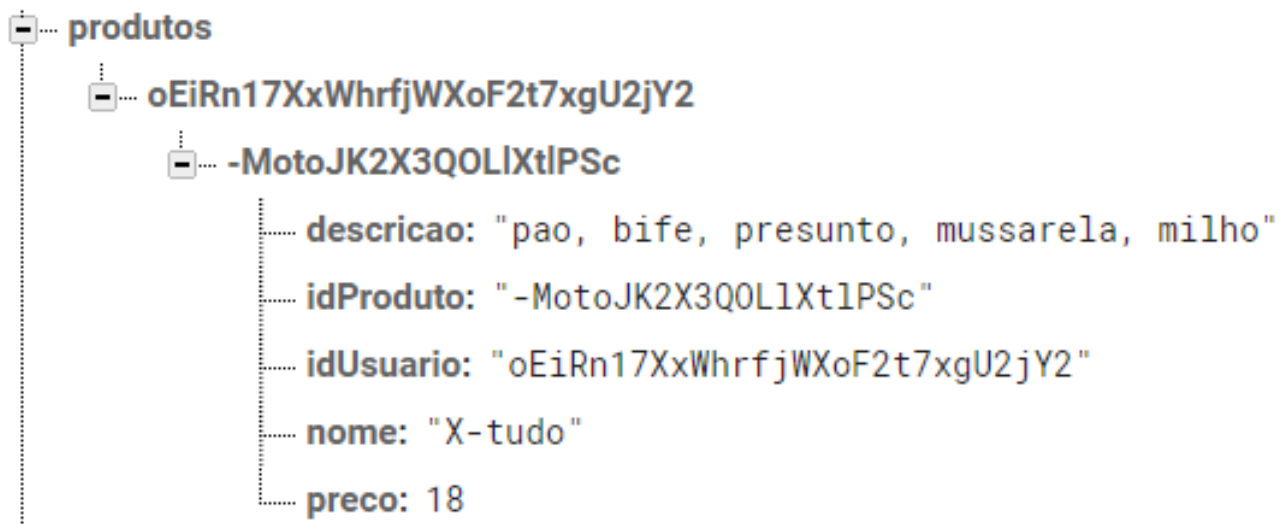


Figura 4.15: Tabela Produtos

A tabela usuários ficará responsável pelo armazenamento das informações dos usuários como nome endereço contato cidade e cep mostrado na figura 4.16.



Figura 4.16: Tabela Usuários

## 4.5 Construção do Aplicativo Móvel

O aplicativo CFFOOD é o aplicativo que tem o intuito de contribuir com o desenvolvimento do sistema de *delivery* na cidade de Córrego Fundo, o aplicativo vem em sua disposição a tela inicial com fundo vermelho e uma imagem de logo como visto na figura 4.17.





Figura 4.17: Tela Splash

A figura 4.17 a tela de apresentação da aplicação chamada *splash* que é exibida na inicialização dando ênfase ao nome do *software*, ela inclui um movimento de entrada ficando exposta por 7 segundos. Para a exibição da imagem foi adicionada uma *imageview*. Logo após será exibida a tela de autenticação.

## 4.6 Tela de login de Empresas e Usuários

Após a exibição da *splash* o aplicativo abre a tela de autenticação como visto na figura 4.18, empresas e usuários precisam realizar a autenticação a fim de confirmar sua identidade digital. Essa validação é realizada utilizando *e-mail* e senha, através da ferramenta “*Authentication*” fornecida pelo Firebase.



Figura 4.18: Tela Login

A tela de autenticação foi desenvolvida com a criação de uma nova *activity* contendo a cor de fundo da tela vermelha, uma *imageview* com logo do aplicativo em seguida um *plaintext* para a inserção do *e-mail* e um *password* para a inserção da senha, a tela tem dois *switchs* informando os meios de configurar cadastros de usuários quanto empresas e por final foi adicionado um *button*, botão de confirmação da autenticação com o nome acessar.

## 4.7 Tela de Login com dados da Empresa

A figura 4.19, mostra tela de *login* com a inserção de dados para a criação de uma empresa, para isso a empresa deverá informar *e-mail* e senha válidos, após inseridos vem a configuração dos *switchs*, para que a empresa seja cadastrada basta ativar o modo cadastrar ativando o *switchs* 1 e depois informar que é uma empresa ativando novamente o *switchs* 2 após serem ativados basta clicar no botão acessar e o cadastro será feito. Após confirmar os dados e acessar será exibida a tela inicial da empresa contendo um aviso.



Figura 4.19: Tela Login

## 4.8 Tela de Aviso

Logo após a autenticação abrirá a tela inicial da empresa com uma mensagem de aviso informando que antes de fazer qualquer configuração, deverá cadastrar seus dados mostrado na figura 4.20.



Figura 4.20: Tela Aviso

Para que essa mensagem seja exibida ao entrar na tela inicial da empresa foi implementada uma *dialog* contendo a mensagem.

## 4.9 Tela Inicial Empresa

Após um clique na mensagem a tela inicial da empresa será exibida, ela será responsável pelos menus de cadastrar empresa, adicionar produtos, exibir os pedidos e sair mostrado na figura 4.21.

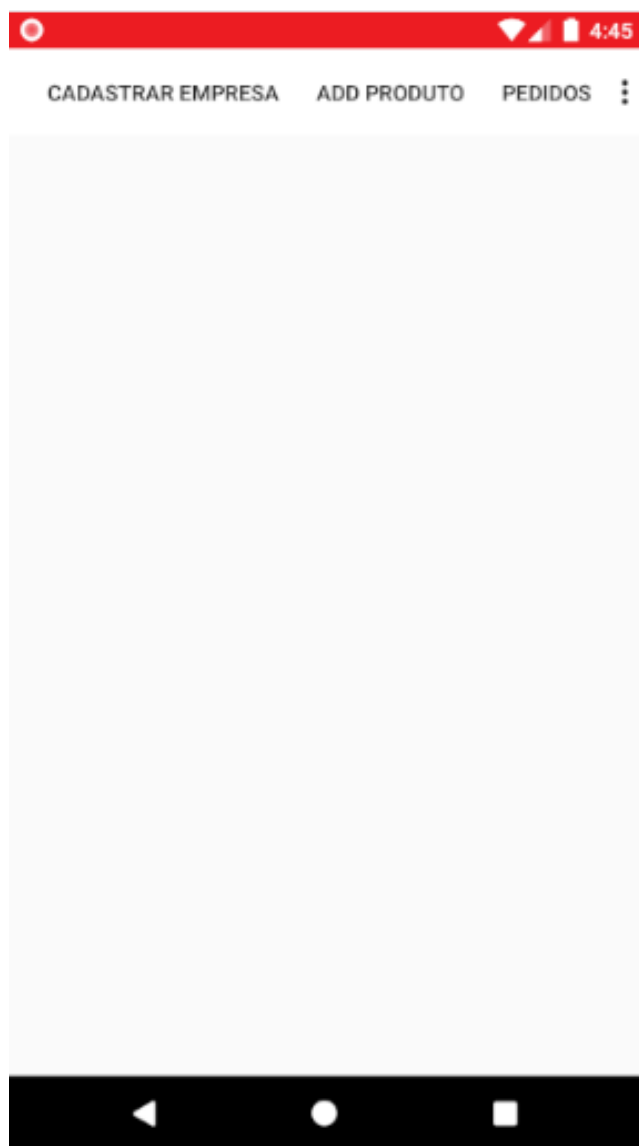


Figura 4.21: Tela Inicial Empresa

Para o desenvolvimento dessa tela foi criada uma *activity* contendo uma, *toolbar* e dentro os menus, feitos com *menuItem* e *menu*. Abaixo será explicado passo a passo de como os menus funcionam e como foram desenvolvidos.

## 4.10 Menu Cadastrar Empresa

No menu cadastrar empresa será feito cadastramento das principais informações de uma empresa como logo, nome, tempo para entrega, taxa a ser cobrada, contato e endereço mostrado na figura 4.22, essas informações são adicionadas no menu do usuário para que possa saber sobre a empresa, mostrado na figura 4.23.



Figura 4.22: Tela Cadastro Empresa

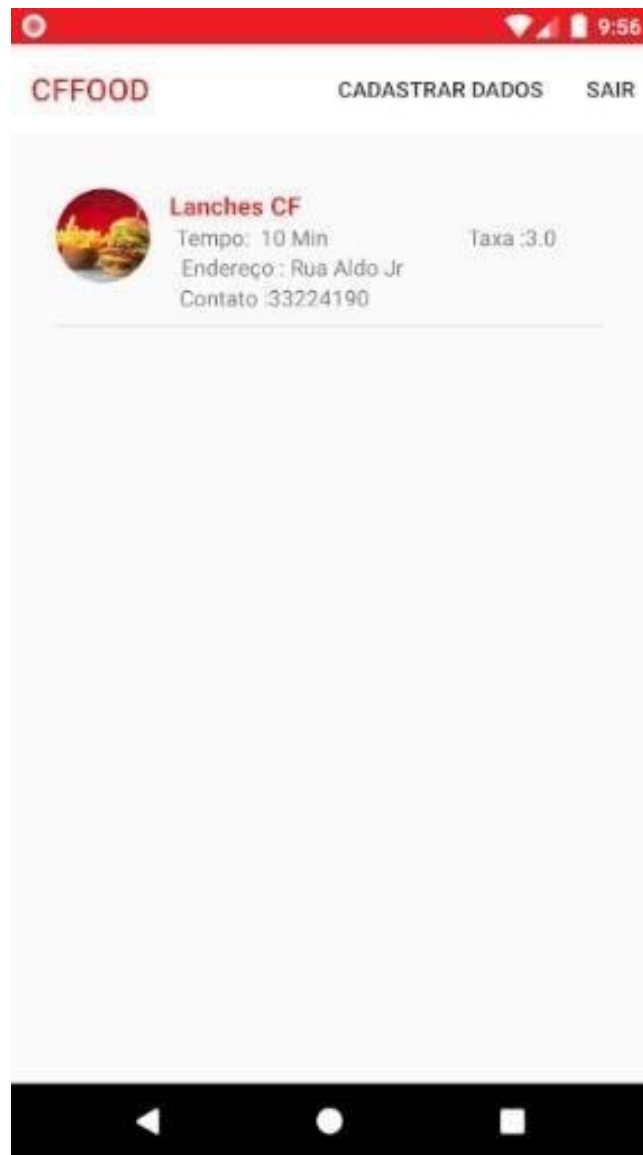


Figura 4.23: Tela Principal do Usuário

Para a criação da tela figura 4.22, foi adicionada uma nova *activity*, com uma, *toolbar* contendo um *textview* com o nome CFFOOD e uma seta para voltar ao menu principal, para definir o logo da empresa foi usado um *imageview*, os campos como nome da empresa, tempo de entrega, taxa, endereço e contato todos feitos com *textview* e por final um *button*, botão para salvar os dados.

## 4.11 Menu Adicionar Produto

O menu adicionar produto foi feito para que a empresa possa adicionar seus produtos e exibi-los aos seus clientes, contendo informações suficientes para definição do produto como nome, descrição e preço, e por final um botão para salvar os dados, como visto na figura 4.24.



Figura 4.24: Tela Cadastro Produto

Para o desenvolvimento da tela foi criado uma nova *activity*, contendo dentro uma, *toolbar* com um *textview* com o nome Novo Produto e uma seta para voltar ao menu principal, os campos nome, descrição e preço foram feitos com *textview* e por final um botão feito com um *button* para salvar os dados.

## 4.12 Relatório de Pedidos

Após a compra de um produto a empresa receberá um relatório contendo informações do usuário e dos produtos que ele comprou no menu Pedidos, a tela contém uma, *toolbar* e dentro dela o nome Pedidos com uma seta ao clicar volta tela principal, para o relatório será exibido uma lista de informações como nome do usuário, endereço, cep, cidade, contato, obs onde o usuário irá informa o valor do pagamento em dinheiro e uma lista dos produtos comprados com seus preços e o valor total do pedido mostrado na figura 4.25.



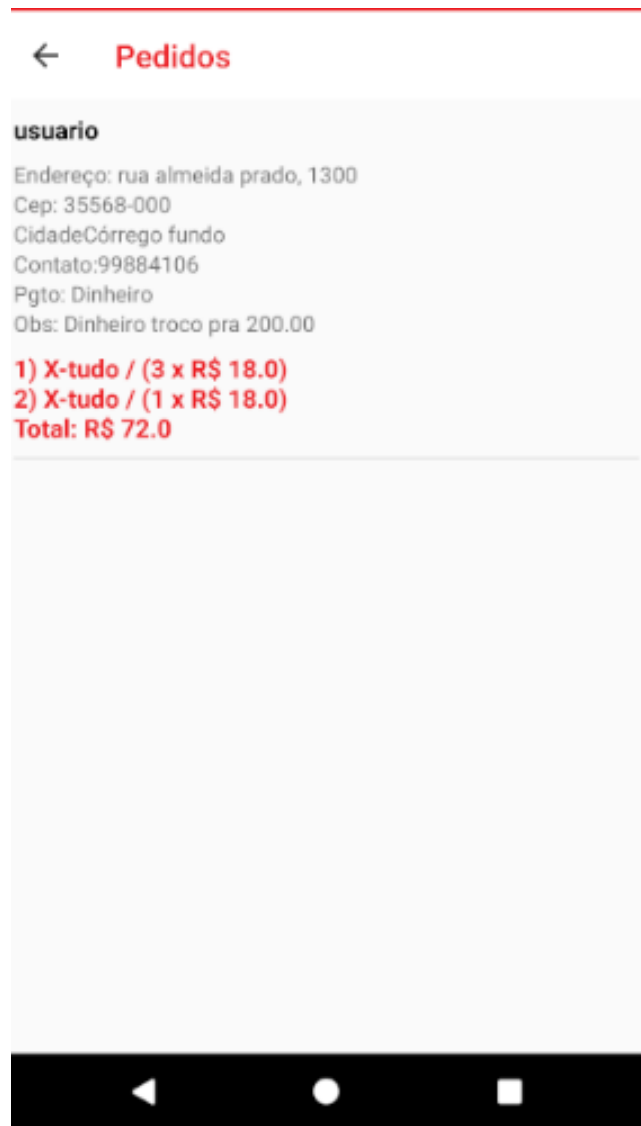


Figura 4.25: Tela Pedidos

Para a exibição dos pedidos foi criado uma nova *activity*, contendo uma *toolbar* com *textView* chamado Pedidos e uma seta para voltar ao menu principal e um divisor abaixo um *recyclerView* que terá os dados nome, endereço, Cep, cidade, contato, forma de pagamento, e obs (observação) e itens de pedidos, feitos com *textView*.

## 4.13 Menu Sair

Ao clicar no menu sair automaticamente sairá da página que você estiver e cairá na tela de *login* novamente, foi criado um menu como mostrado na figura 4.26.

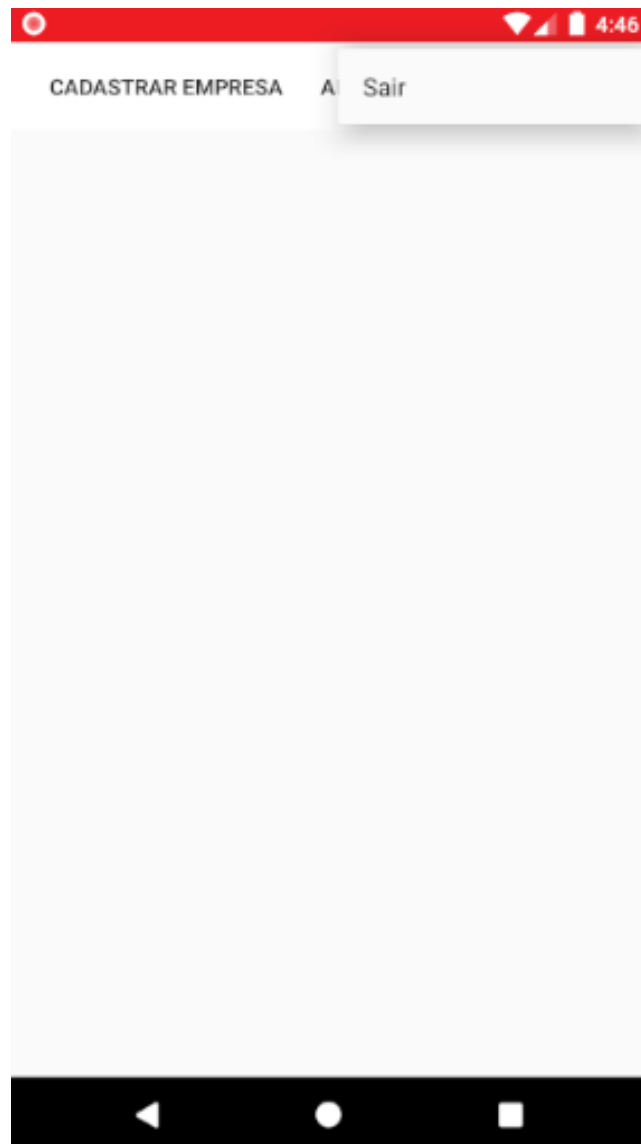


Figura 4.26: Tela Principal

Como mostrando anteriormente todo o processo que uma empresa precisa para cadastrar, exibir seus produtos e visualizar os pedidos, agora será exibido do o processo para o usuário.

## 4.14 Tela Autenticação Usuário

A figura 4.27 mostra a tela de *login* com os dados de um usuário informando *e-mail* e senha e configurando os *switchs* no modo cadastrar e usuário após a inserção dos dados o usuário será cadastrado automaticamente no aplicativo.



Figura 4.27: Tela Login com Dados do Usuário

Após confirmar os dados e acessar será exibida a tela inicial da empresa contendo um aviso.

## 4.15 Tela de Aviso

Logo após a autenticação abrirá a tela inicial do usuário com uma mensagem de aviso informando que antes de fazer qualquer configuração, deverá cadastrar seus dados mostrado na figura 4.28.

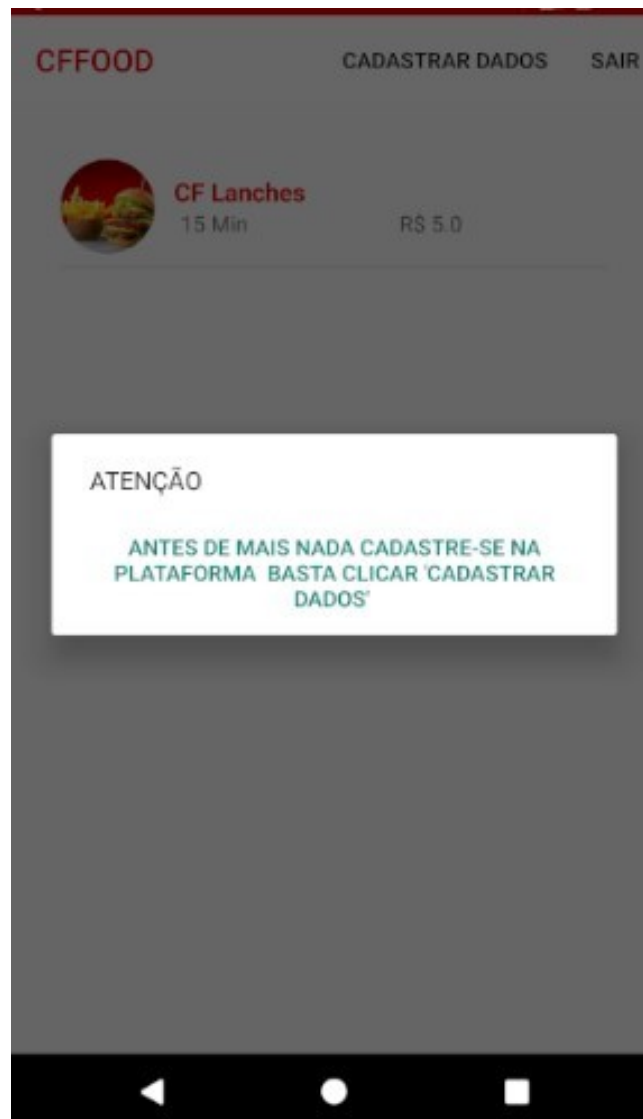


Figura 4.28: Tela Aviso

Para que essa mensagem seja exibida ao entrar na tela inicial da empresa foi implementada uma *dialog* contendo uma mensagem.

## 4.16 Tela inicial Usuário

Após um clique na mensagem abrirá a tela inicial do usuário, onde o usuário poderá cadastrar seus dados, visualizar empresas e sair. A tela contém uma *toolbar* e dentro um texto com nome do aplicativo e os menus cadastrar usuário, sair e uma lista das empresas cadastradas mostrado na figura 4.29.

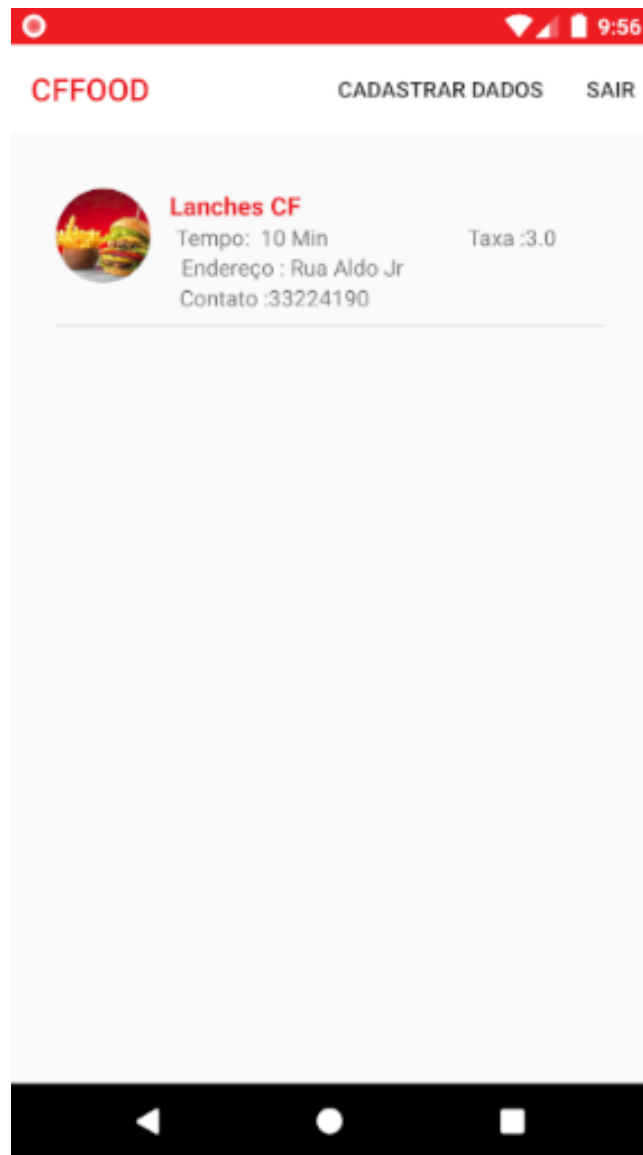


Figura 4.29: Tela Inicial do Usuário

Para a criação foi feita com uma nova *activity* contendo *toolbar* e dentro os menus cadastrar dados e sair, feitos com *menuItem*, e um divisor, para a exibição das empresas foi adicionado um *recyclerview* e dentro contendo uma imagem (logo da empresa) feita por uma *imageView* e os campos tempo, endereço, contato e taxa feitos por *textView* e no final um divisor. Abaixo será explicado passo a passo de como os menus funcionam e como foram desenvolvidos.

## 4.17 Cadastrar Dados do Usuário

No menu cadastrar dados do usuário será feito cadastramento das principais informações de um usuário como nome, endereço, Cep, cidade, e número de contato mostrado na figura 4.30, essas informações são adicionadas no relatório da compra dos produtos, para que a empresa possa saber onde entregar, e por final um botão para salvar as informações.



Figura 4.30: Tela de Cadastro do Usuário

Para a tela do menu cadastrar dados do usuário foi criada uma *activity*, com uma *toolbar* dentro dela uma seta para voltar e abaixo os campos nome, endereço, Cep, cidade, número de contato criados com *textview* e um *button*, um botão para salvar os dados.

## 4.18 Menu Sair

Ao clicar no menu sair, automaticamente sairá da página que você estiver e cairá na tela de *login* novamente. Após todo o procedimento de cadastro do usuário na plataforma for concluído o usuário poderá efetuar suas compras no aplicativo.

## 4.19 Efetuar Compra

Após realizar o cadastro, o usuário poderá fazer suas compras no aplicativo, bastando clicar na empresa desejada exemplo mostrado na figura 4.31, ao clicar irá aparecer todos os produtos cadastrados da empresa e suas principais informações como nome, descrição e preço mostrado na figura 4.32.

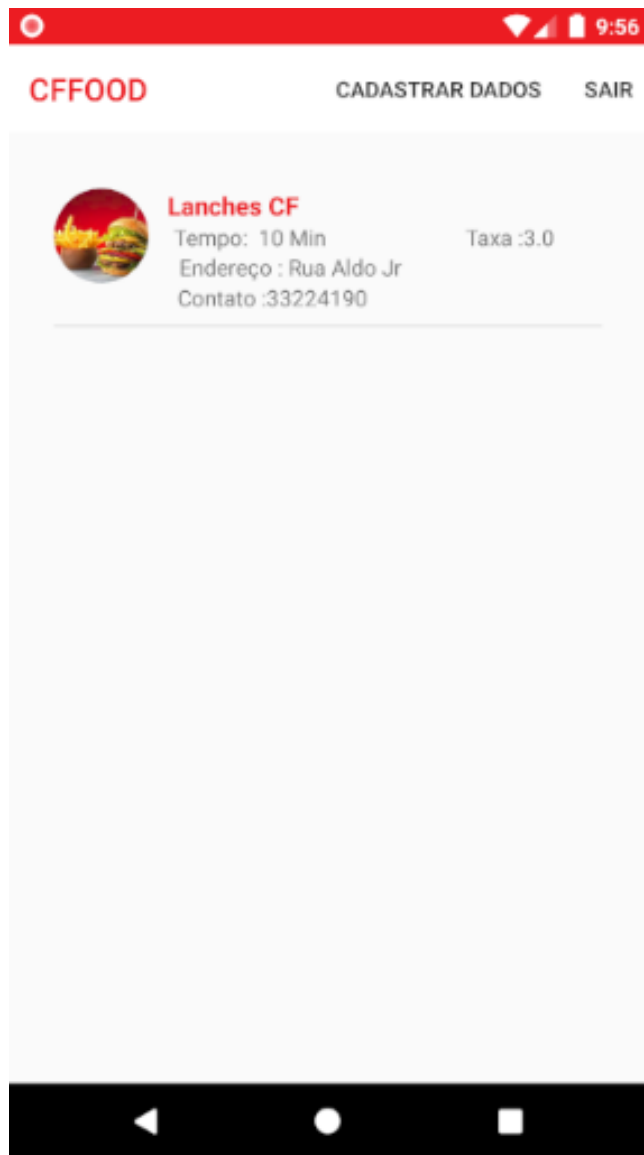


Figura 4.31: Tela Principal do Usuário



Figura 4.32: Tela do Usuário Produtos

Para o desenvolvimento da tela 4.32 foi criada uma *activity* contendo uma *toolbar* e dentro um *textview* com a palavra compra e um menu confirmar pedido feito com *menuItem*, foi adicionado um *linearlayout vertical* e dentro contendo *linearlayout horizontal* com logo da empresa, feito com *imageview* e *textview* para o nome da empresa, foi criado outro *linearlayout vertical* contendo dentro *imageview* com logo de um carrinho e dois *textview* como nomes quantidade e valor e por final um *recyclerview* onde vai conter todos os produtos e suas informações feitos com *textview*.

## 4.20 Tela Informar a Quantidade do produto

Após clicar no produto desejado irá abrir uma *dialog* com uma mensagem perguntando a quantidade de itens que deseja comprar, basta informar a quantidade e confirmar mostrado na figura abaixo 4.33





Figura 4.33: Tela Informar a Quantidade

Após a inserção da quantidade de produtos, o carrinho exibirá a quantidade de itens e o preço total a ser pago exibido na imagem 4.34.



Figura 4.34: Tela Produtos

## 4.21 Tela Mensagem Pagamento

Depois é a etapa de confirmar pedido basta clicar no menu confirmar pedido, assim abrirá uma a mensagem, onde o usuário irá informar o valor que ele irá pagar pelos produtos e confirmar após esse procedimento está feito a compra no aplicativo mostrado na figura 4.35.



Figura 4.35: Tela Informar Pagamento

---

## Conclusão

---

A pandemia do covid-19 causou diversos impactos no mundo todo, onde o isolamento social foi uma alternativa de conter a doença, sem a presença de públicos, empresas do ramo alimentício tiveram que se reinventar colocando como uma válvula de escape aplicativos de *delivery* onde as empresas possam exibir seus produtos e os usuários a fazerem suas compras. Este trabalho teve como objetivo desenvolver e documentar um aplicativo de *delivery*, para a cidade de Córrego Fundo. Ao finalizar o desenvolvimento da aplicação desde o banco de dados até o aplicativo, foi possível perceber o quanto o desenvolvimento para aplicações móveis pode contribuir comércio com *delivery* em cidades pequenas. São mostradas as etapas e a documentação necessária para o desenvolvimento do aplicativo proposto, do levantamento de requisitos a implementação, juntamente com as tecnologias usadas para a construção do aplicativo, visto que o ele permite visualização das empresas e informações individuais de cada produto, além de permitir que os clientes façam suas compras.

### 5.1 Trabalhos Futuros

Este trabalho apresenta um aplicativo de *delivery* genérico que busca auxiliar no intermédio da venda de produtos. Todavia, apenas as principais funcionalidades foram desenvolvidas. Sabendo disso, trabalhos futuros podem ser incorporados no aplicativo CFFOOD para melhor experiência de seus usuários, como por exemplo:

- Publicar na *Play Store*
- Desenvolver um módulo vinculado a api para pagamento via cartão de crédito
- Melhorias na Interface.
- Expansão para aplicação Híbrida.



## Referências Bibliográficas

---

AUGUSTO, G. *Java: tudo o que você precisa saber para começar*. 2021. Disponível em: <<https://www.zup.com.br/blog/java>>.

BULL, P. *Como Organizar um Serviço de Delivery eficiente.pdf*. 2020. Disponível em: <<https://www.consumidormoderno.com.br/2020/07/08/gastos-com-delivery-crescem-mais-de-94-durante-a-pandemia>>.

CAMPOS, A. C. *Cai para 33,8 milhões número de pessoas rigorosamente isoladas Cerca de 5,1 milhões de pessoas interromperam o isolamento*. 2020. Disponível em: <<https://agenciabrasil.ebc.com.br/geral/noticia/2020-10/cai-para-338-milhoes-numero-de-pessoas-rigorosamente-isoladas>>.

CANALTECH. *Ifood*. 2011. Disponível em: <<https://canaltech.com.br/empresa/ifood/>>.

DAU, G. *O aumento dos gastos em aplicativos de Delivery com a pandemia*. 2021. Disponível em: <<https://www.jornalcontabil.com.br/o-aumento-dos-gastos-em-aplicativos-de-delivery-com-a-pandemia/>>.

DEVELOPERS. *Conheça o Android Studio*. 2021. Disponível em: <<https://developer.android.com/studio/intro?hl=pt-br>>.

EL-KASSAS, W. S. e. a. *Taxonomy of cross-platform mobile applications development approaches*. 2015.

FIREBASE. *Firestore Documentation*. 2021. Disponível em: <<https://firebase.google.com/docs>>.

FONSECA, K. d. O. *Aplicativo crowdsourcing para localização de serviços especializados em bem-estar animal: guipet*. Universidade do Vale do Rio dos Sinos., 2015.

FRANÇA, A. *8 em cada 10 estabelecimentos comerciais manterão delivery após pandemia, diz estudo*. 2020. Disponível em: <<https://istoe.com.br/8-em-cada-10-estabelecimentos-comerciais-manterao-delivery-apos-pandemia-diz-estudo/>>.

FRANÇA, I. *Investimentos: tendências para o mercado de delivery no Brasil e no Mundo em 2020*. 2020. Disponível em: <<https://blog.deliverymuch.com.br/mercado-de-delivery-no-brasil/>>.

GARBIN, M. *Sistema de recomendação mobile: um estudo de caso para delivery*. Universidade Federal de Santa Maria, 2014.

- GORDON, J. *Impacto da pandemia na escolha dos consumidores pelo delivery*. 2021. Disponível em: <<https://www.ecommercebrasil.com.br/artigos/impacto-da-pandemia-na-escolha-dos-consumidores-pelo-delivery/>>.
- GRANDI, G. *Apesar da crise, mais de 437 mil empresas de alimentação abriram no Brasil em 2020*. 2021. Disponível em: <<https://www.gazetadopovo.com.br/bomgourmet/negocios-e-franquias/novas-empresas-alimentacao-brasil/>>.
- GUIMARAES, A. C. *Pesquisa mostra que crescimento de delivery continuará mesmo depois da pandemia*. 2020. Disponível em: <<https://blogs.oglobo.globo.com/ancelmo/post/pesquisa-mostra-que-crescimento-de-delivery-continuara-mesmo-depois-da-pandemia.html>>.
- IBGE. *Córrego Fundo*. 2021. Disponível em: <<https://cidades.ibge.gov.br/brasil/mg/corrego-fundo/panorama>>.
- IMPRESA. *Delivery cresce 60% durante a pandemia e segmento de alimentação lucra com esse crescimento*. 2021. Disponível em: <<https://mapadasfranquias.com.br/noticia/delivery-cresce-60-durante-a-pandemia-e-segmento-de-alimentacao-lucra-com-esse-crescimento/>>.
- JUNIOR, F. *Delivery transformou tendência em necessidade e continua em crescimento*. 2021. Disponível em: <<https://jornal.usp.br/atualidades/delivery-transformou-tendencia-em-necessidade-e-continua-em-crescimento/>>.
- JUNIOR, S. A. F. Protótipo de um aplicativo android para pedidos de lanches e um portal web para gestão e monitoramento. *REPOSITÓRIO DE RELATÓRIOS-Sistemas de Informação*, n. 1, 2015.
- MALAVOLTA, I. *Além dos aplicativos nativos: tecnologias da web para o resgate!* 2016. Disponível em: <<https://dl.acm.org/doi/abs/10.1145/3001854.3001863>>.
- MATOS, B. R. D.; BRITTO, J. G. de. Estudo comparativo entre o desenvolvimento de aplicativos móveis utilizando plataformas nativas e multiplataforma. 2016.
- MELO, C. F. d. A. et al. Bom delivery: um aplicativo de delivery genérico para cidades do interior. Universidade Federal de Campina Grande, 2019.
- MORIBE, F. *Firebase — Vantagens de um BaaS para sua Startup*. 2016. Disponível em: <<https://medium.com/@fgmoribe/firebase-vantagens-de-um-baas-para-sua-startup-38fd3891329a>>.
- ORACLE. *Java e você, faça download hoje*. 2021. Disponível em: <<http://docs.oracle.com>>.
- QUIRINO, L. d. S. *REDE DE TELEMARKETING: NÍVEL DE SATISFAÇÃO DOS CLIENTES COM OS SERVIÇOS DE ENTREGA A DELIVERY*. Dissertação (B.S. thesis), 2016.
- RIGOLO, J. *STARTUP QUER SER O IFOOD PARA AS CIDADES PEQUENAS*. 2019. Disponível em: <<https://inovaprudente.com.br/noticias/startup-quer-ser-o-ifood-para-as-cidades-pequenas.html>>.
- ROBISON. *Desenvolvimento Android utilizando a IDE Android Studio*. 2015. Disponível em: <<https://www.devmedia.com.br/desenvolvimento-android-utilizando-a-ide-android-studio/33872>>.
- SILVA, M. R. G. da et al. O crescimento das empresas de delivery no contexto da pandemia. Universidade Federal de Minas Gerais, 2021.

SOARES, A. C. N.; LIMA, M. R. da S. Serviços de delivery alimentício e suas precauções em tempos da pandemia de sars-cov-2 (covid-19). *Brazilian Journal of Health Review*, v. 3, n. 3, p. 4217–4226, 2020.

TECHTUDO. *O que é iFood? Saiba como funciona o app de delivery*. 2016. Disponível em: <<https://www.techtudo.com.br/noticias/2016/07/o-que-e-ifood-saiba-como-funciona-o-app-de-delivery.ghtml>>.

TIAGO. *Aplicativos Nativos: Quando Vale A Pena E Como Criar O Seu*. 2021. Disponível em: <<https://mundodevops.com/blog/aplicativos-nativos/>>.