

**UNIVERSIDADE FEDERAL DE SERGIPE
CAMPUS ALBERTO CARVALHO
DEPARTAMENTO DE SISTEMAS DE INFORMAÇÃO**

RAFAEL ALVES CÔRTEZ

**INTEGRAÇÃO DE SISTEMAS UTILIZANDO API REST E
GEOLOCALIZAÇÃO VISANDO A BUSCA PELO MENOR
PREÇO NO COMÉRCIO VAREJISTA**

**ITABAIANA
2021**

**UNIVERSIDADE FEDERAL DE SERGIPE
CAMPUS ALBERTO CARVALHO
DEPARTAMENTO DE SISTEMAS DE INFORMAÇÃO**

RAFAEL ALVES CÔRTEZ

**INTEGRAÇÃO DE SISTEMAS UTILIZANDO API REST E
GEOLOCALIZAÇÃO VISANDO A BUSCA PELO MENOR
PREÇO NO COMÉRCIO VAREJISTA**

Trabalho de Conclusão de Curso
submetido ao Departamento de
Sistemas de Informação da
Universidade Federal de Sergipe
como requisito parcial para a
obtenção do título de Bacharel em
Sistemas de Informação.

Orientador: Dr. JOSÉ AÉLIO DE OLIVEIRA JÚNIOR

**ITABAIANA
2021**

CÔRTEZ, Rafael Alves.

Integração de sistemas utilizando API REST e geolocalização visando a busca pelo menor preço no comércio varejista / Rafael Alves Côrtes – Itabaiana: UFS, 2021. 117f.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal de Sergipe, Curso de Sistemas de Informação, 2021.

1. Sistema *Delivery* de produtos.
 2. Desenvolvimento de Software.
 3. Sistemas de Informação.
- I. Integração de sistemas utilizando API REST e geolocalização visando a busca pelo menor preço no comércio varejista.

RAFAEL ALVES CÔRTEZ

**INTEGRAÇÃO DE SISTEMAS UTILIZANDO API REST E
GEOLOCALIZAÇÃO VISANDO A BUSCA PELO MENOR
PREÇO NO COMÉRCIO VAREJISTA**

Trabalho de Conclusão de Curso submetido ao corpo docente do Departamento de Sistemas de Informação da Universidade Federal de Sergipe (DSIITA/UFS) como parte dos requisitos para obtenção do grau de Bacharel em Sistemas de Informação.

Itabaiana, 20 de dezembro de 2021 da aprovação.

BANCA EXAMINADORA:

**Prof(a) José Aélío de Oliveira Júnior, Doutor
Orientador
DSIITA/UFS**

**Prof(a) André Vinicius Rodrigues Passos Nascimento,
Mestre
DSIITA/UFS**

**Prof(a) Methanias Colaço Rodrigues Júnior, Doutor
DSIITA/UFS**

DEDICATÓRIA

Dedico este trabalho a minha querida mãe, Jeane, ela nos deixou durante a pandemia em decorrência das complicações da COVID-19 e foi um acontecimento que ninguém esperava. Onde estiver mãe, te agradeço por tudo. Este trabalho, é um dos resultados dos seus esforços, com sua garra e dedicação me ajudou em todos os momentos da minha vida.

AGRADECIMENTOS

Agradeço a minha querida mãe Jeane que não está mais nesse plano, uma mulher guerreira, sempre me deu amor e ternura e mesmo não tendo as melhores condições financeiras, fez todos os esforços para não afetar a vida escolar dos seus filhos, e que nos deixou esse ano durante a pandemia, a meu pai Reginaldo sempre trabalhando para ajudar a todos e que sempre me incentivou e fez todos os esforços para que eu continuasse os estudos, me ensinou os valores humanos que sigo até hoje.

Minha esposa Raphaela que sempre me apoiou em todos os momentos a não desistir e finalizar o curso, e mesmo mais cansada que eu, apesar de todas as adversidades, sempre estudando e mostrando sua garra para que eu não desistisse. A minha filha Raphaele que me fez perseverar para alcançar novos objetivos por ela e me traz a esperança de um mundo melhor.

Ao meu orientador, prof. Dr. José Aélío de Oliveira Júnior, por me ajudar nessa jornada durante o trabalho, em períodos conturbados se preocupou e me ajudou em todos os momentos que precisei, muito obrigado.

Prof. Dr. Methanias Colaço Rodrigues Júnior e Prof. Msc. André Vinicius Rodrigues Passos, agradeço por contribuir para a conclusão desse trabalho e também durante o curso, seus ensinamentos não serão esquecidos.

Obrigado a todos os meus amigos que conheci durante esses anos de curso em especial Luiz, Anailson, Clóvis, Alcymar, Jéssica, Suellington, Cléverton e a todos com quem estudei ou compartilhei conhecimento, além de Adilton claro, que sempre me ajudou quando foi preciso, principalmente nesses últimos anos com a pandemia. Meu muito obrigado, muitos ainda mantenho contato diariamente e todos estão progredindo nas suas carreiras de forma satisfatória, estou muito feliz por todos vocês.

Agradeço a todos do Departamento de Sistemas de Informação do Campus Itabaiana. E a todos que me apoiaram e acreditaram em mim durante esses anos e me deram forças quando eu não acreditei, meu muito obrigado.

CÔRTEZ, Rafael Alves. **Integração de Sistemas Utilizando API REST e Geolocalização Visando a Busca Pelo Menor Preço no Comércio Varejista**. 2021. Trabalho de Conclusão de Curso – Curso de Sistemas de Informação, Departamento de Sistemas de Informação, Universidade Federal de Sergipe, Itabaiana, 2021.

RESUMO

Com o avanço da tecnologia, as pessoas estão buscando mais comodidade ao usufruir das novas ferramentas que surgem com o propósito de ajudar nas tarefas do cotidiano. Nos últimos anos surgiram vários aplicativos de delivery de produtos, e a utilização desses cresceu não apenas pela facilidade, mas também pelas restrições causadas pelas adversidades do COVID-19. Apesar da conveniência, a maioria dos aplicativos não possuem uma logística adequada em relação a compras que podem ter produtos distribuídos em várias lojas, ademais, a busca pelo menor preço de um mesmo produto não é funcional o que pode tornar exaustivo economizar ao comprar os produtos mais baratos nas lojas disponíveis pelo aplicativo. O presente trabalho tem como objetivo o desenvolvimento de um sistema no qual o cliente consiga realizar suas compras nas lojas da sua região e recebê-las no endereço indicado de forma integral, diferente das entregas fracionadas, que normalmente, é realizado uma entrega por loja. Além disso, a busca pelo menor preço de um mesmo produto é facilitada, pois um mesmo produto pode ser consultado em todas as lojas disponíveis, e listado do menor para o maior preço.

Palavras-chave: Delivery, Comodidade, Busca Pelo Menor Preço.

ABSTRACT

With the advances in technology, people are looking for more convenience in using the new tools that are emerging with the purpose of helping in daily tasks. Over the last few years several Product Delivery Apps have emerged, and their use has grown not only because of their convenience, but also because of the restrictions caused by the COVID-19 adversities. Despite the conveniences, most of the applications do not have an adequate logistics, especially when it comes to purchases which may have products distributed in several stores. Furthermore, the searching for the lowest price of the same product is not functional, which can become exhaustive when purchasing the cheapest products from the stores provided by the application. This work aims to develop a system in which customers can make their purchases from the stores in their area and have them delivered to the address specified in a complete way, differently from the fragmented deliveries, which usually is a delivery by each store. Additionally, searching for the cheapest price of the same product is easier, because the same product can be checked out in all available stores, and be listed from lowest to highest price.

Key-words: *Delivery, Convenience, Search for the lowest price.*

LISTA DE FIGURAS

Figura 1 – Spring Framework com Java Web Container (BOAGLIO, 2017).....	28
Figura 2 – Spring Boot no controle total da aplicação web (BOAGLIO, 2017).....	28
Figura 3 – Arquitetura do Xamarin (XAMARIN, 2021)	33
Figura 4 – Diagrama de casos de uso da aplicação do cliente	38
Figura 5 – Diagrama de casos de uso da aplicação do entregador	48
Figura 6 – Diagrama de casos de uso da aplicação do lojista	56
Figura 7 – Esquema de dados do sistema.....	65
Figura 8 – Tela inicial do aplicativo sem efetuar o login.....	74
Figura 9 – Mensagem para informar ao usuário que o login precisa ser efetuado.....	74
Figura 10 – Tela inicial do aplicativo com login realizado	75
Figura 11 – Menu lateral sem efetuar o login.....	76
Figura 12 – Menu lateral com login realizado.....	76
Figura 13 – Tela de autenticação.....	77
Figura 14 – Tela de cadastro	78
Figura 15 – Pop-up de pesquisa	79
Figura 16 – Tela de mudança de endereço de entrega.....	80
Figura 17 – Mensagem de cep não encontrado	80
Figura 18 – Tela de mudança de senha	81
Figura 19 – Mensagens de erro ao tentar mudar a senha	82
Figura 20 – Tela para adicionar produto ao carrinho	83
Figura 21 – <i>Pop-up</i> para adicionar produto ao carrinho.....	84
Figura 22 – Tela de carrinho de compras	85
Figura 23 – Mensagem de confirmação para remover produto do carrinho	85
Figura 24 – <i>Pop-up</i> para edição do produto no carrinho de compras	86
Figura 25 – Tela de histórico de compras	87
Figura 26 – Opções da compra no histórico de compras.....	87
Figura 27 – Tela de detalhes da compra.....	88
Figura 28 – Tela de rastrear entrega	89
Figura 29 – Tela inicial de login do aplicativo.....	90
Figura 30 – Mensagem de erro ao logar.....	90
Figura 31 – Tela de cadastro de usuário.....	91
Figura 32 – Menu lateral do aplicativo.....	92
Figura 33 – Tela de paradas por loja	93
Figura 34 – Pop-up com opções de item da lista de paradas por loja	93
Figura 35 – Tela de compras por loja.....	94
Figura 36 – Tela de detalhe da compra	95
Figura 37 – Tela de compras já coletadas	96
Figura 38 – <i>Pop-up</i> de compra já coletada.....	96
Figura 39 – Tela de detalhamento de compra já coletada	97
Figura 40 – Tela rastrear endereço da loja	98
Figura 41 – Tela rastrear endereço do cliente	99
Figura 42 – Tela inicial de autenticação.....	100
Figura 43 – Tela de cadastro de usuário.....	101
Figura 44 – Menu lateral da aplicação do lojista.....	102

Figura 45 – Tela de compras a coletar.....	103
Figura 46 – Tela de Detalhamento de compra para coletar.....	104
Figura 47 – Tela de lista de produtos cadastrados na loja.....	105
Figura 48 – Opções de produto da lista.....	105
Figura 49 – Tela de lista de Adição de produtos na loja.....	106
Figura 50 – Tela de Alteração de produtos.....	107
Figura 51 – Alteração de estoque de produto.....	108
Figura 52 – Pesquisa de produtos da loja.....	109
Figura 53 – Tela para alterar senha.....	110
Figura 54 – Tela de lista de compras já coletadas.....	111
Figura 55 – Tela de detalhamento de compra já coletada.....	112

LISTA DE QUADROS

Quadro 1 - Requisitos funcionais e não funcionais da ferramenta.....	37
Quadro 2 - Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.....	40
Quadro 3 - Descrição do Caso de Uso em Nível de Sistema – Autenticação	41
Quadro 4 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Adicionar produto ao carrinho	41
Quadro 5 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Remover produto ao carrinho.....	42
Quadro 6 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Alterar produto ao carrinho	42
Quadro 7 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Finalizar Compra	43
Quadro 8 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Detalhes Compra	44
Quadro 9 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Recriar Compra	44
Quadro 10 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Rastrear Entrega.....	45
Quadro 11 - Descrição do Caso de Uso em Nível de Sistema - Alterar Senha.....	46
Quadro 12 - Descrição do Caso de Uso em Nível de Sistema - Alterar Endereço de Entrega	46
Quadro 13 - Requisitos funcionais e não funcionais da aplicação.....	47
Quadro 14 - Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.....	49
Quadro 15 - Descrição do Caso de Uso em Nível de Sistema – Autenticação	50
Quadro 16 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Listar compras a coletar por loja.....	51
Quadro 17 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Ver localização da loja no mapa.....	51
Quadro 18 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Concluir Recebimento	52
Quadro 19 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Verificar Detalhes da compra já coletada	52
Quadro 20 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Visualizar Endereço de Entrega	53
Quadro 21 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Finalizar entrega	54
Quadro 22 - Descrição do Caso de Uso em Nível de Sistema - Alterar Senha.....	54
Quadro 23 - Requisitos funcionais e não funcionais da aplicação.....	55
Quadro 24 - Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.....	57
Quadro 25 - Descrição do Caso de Uso em Nível de Sistema – Autenticação	58
Quadro 26 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Listar Compras efetuadas pelos clientes na loja	58
Quadro 27 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Verificar detalhes da compra	59
Quadro 28 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Concluir Coleta.....	59

Quadro 29 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Listar Compras efetuadas pelos clientes na loja já coletadas	60
Quadro 30 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Listar produtos cadastrados da loja	60
Quadro 31 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Cadastrar Produtos	61
Quadro 32 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Editar Produtos	62
Quadro 33 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Remover produtos	63
Quadro 34 - Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Editar Estoque dos produtos	64
Quadro 35 - Descrição do Caso de Uso em Nível de Sistema - Alterar Senha	64
Quadro 36 - Definição das tabelas	68
Quadro 37 - Definição dos atributos	73

SUMÁRIO

1	INTRODUÇÃO	17
1.1	Objetivos	19
1.1.1	Geral	19
1.1.2	Específicos	19
1.2	Relevância do trabalho	20
1.3	Metodologia	20
1.4	Estrutura do trabalho	21
2	REVISÃO BIBLIOGRÁFICA	22
2.1	SOA	22
2.1.1	SOAP	23
2.1.2	WSDL	24
2.1.3	UDDI	24
2.1.4	REST	25
2.1.4.1	Spring Boot	26
2.2	Geolocalização	28
2.2.1	GPS	28
2.2.2	GPS	29
2.3	Desenvolvimento Mobile	30
2.3.1	Aplicações Híbridas	30
2.3.2	Desenvolvimento Multiplataforma	31
2.3.2.1	Xamarin	31
2.4	Trabalhos Relacionados	32
2.4.1	FASTLINE: Aplicativo mobile para lanchonetes	32
2.4.2	Uber Eats	33
2.4.3	Extra Delivery	34
2.4.4	Quero Delivery	34
2.4.5	iFood	34
2.4.6	Bom Delivery: Um aplicativo de delivery genérico para cidades do interior	35

3 PROJETO BUSCOU BARATO	36
3.1 Buscou Barato Módulo Cliente	36
3.1.1 Requisitos funcionais e não funcionais	36
3.1.2 Casos de Uso.....	37
3.1.2.1 Casos de Uso – Nível de Usuário	38
3.1.2.2 Casos de Uso – Nível de Sistema	39
3.1.2.2.1 Caso de Uso Cadastrar Usuário	39
3.1.2.2.2 Caso de Uso Autenticação.....	40
3.1.2.2.3 Caso de Uso Gerenciar Produtos do Carrinho	41
3.1.2.2.4 Caso de Uso Gerenciar Compras	43
3.1.2.2.5 Caso de Uso Alterar Senha	45
3.1.2.2.6 Caso de Uso Alterar Endereço de Entrega.....	46
3.2 Buscou Barato Módulo Entregador.....	46
3.2.1 Requisitos funcionais e não funcionais	46
3.2.2 Casos de Uso.....	47
3.2.2.1 Casos de Uso – Nível de Usuário	48
3.2.2.2 Casos de Uso – Nível de Sistema	49
3.2.2.2.1 Caso de Uso Cadastrar Usuário	49
3.2.2.2.2 Caso de Uso Autenticação.....	50
3.2.2.2.3 Caso de Uso Gerenciar Entregas.....	50
3.2.2.2.4 Caso de Uso Alterar Senha	54
3.3 Buscou Barato Módulo Lojista.....	55
3.3.1 Requisitos funcionais e não funcionais	55
3.3.2 Casos de Uso.....	56
3.3.2.1 Casos de Uso – Nível de Usuário	56
3.3.2.2 Casos de Uso – Nível de Sistema	57
3.3.2.2.1 Caso de Uso Cadastrar Usuário	57
3.3.2.2.2 Caso de Uso Autenticação.....	58
3.2.2.2.3 Caso de Uso Gerenciar Coletas	58
3.2.2.2.4 Caso de Uso Gerenciar Produtos	60
3.2.2.2.5 Caso de Uso Alterar Senha	64
3.4 Modelagem de Dados	65
3.4.1 Componentes do modelo de dados	66

4	BUSCOU BARATO	73
4.1	Buscou Barato Cliente.....	73
4.2	Buscou Barato Entregador	89
4.3	Buscou Barato Lojista	99
5	CONCLUSÃO E TRABALHOS FUTUROS.....	113
	REFERÊNCIAS	114

1 INTRODUÇÃO

A facilidade e a comodidade para efetuar tarefas simples do cotidiano tem aumentado com o avanço da tecnologia nos últimos anos. As pessoas estão buscando meios que lhes proporcionem uma forma de concluir seus afazeres de forma simples, rápida e com menos esforço a todo tempo, seja criando novos meios ou usando outros já criados. Muitos produtos foram desenvolvidos para suprir essa necessidade da população, desde os primeiros relógios solares para facilitar a leitura da hora diária até dispositivos avançados como os celulares atuais, utilizados para uma infinidade de tarefas.

Com todo o avanço tecnológico, o desenvolvimento de software mudou consideravelmente ao longo dos anos, os fornecedores precisam gerar soluções cada vez mais integradas e comunicativas para satisfazer a necessidade do mercado competitivo atual, visto que os clientes precisam de softwares capazes de se comunicar com diversos tipos de dispositivos diferentes. Segundo GUINARD *et al.* (2010) as causas da diversidade de dispositivos é o hardware menor, mais barato e poderoso, e estes utilizam tecnologias de integração para se comunicarem.

De acordo com MEIRELLES (2021), coordenador da 32ª Pesquisa Anual do Uso de Tecnologia da Informação A evolução e tendência dos computadores (de mesa/desktops, portáteis/notebooks e tablets) em uso no Brasil é de 200 milhões em 2021. Uma densidade *per capita* de 94%, acima da média mundial de 82%. Já para *smartphones* o número é ainda maior, são 242 milhões de dispositivos. Por conseguinte, os desenvolvedores de softwares devem levar estes dados em conta, pois produzir um sistema que alcance todo o público alvo, certamente não irá abranger apenas um tipo de plataforma, e este software precisa comunicar-se com o resto do sistema. A partir dessa necessidade surgiram vários padrões para ajudar nesta integração, sendo a Arquitetura Orientada a Serviços (SOA) uma delas.

Os *smartphones* são os principais alvos de compras dos consumidores, e o sistema operacional mais utilizado nestes dispositivos é o Android. De acordo com a COMTECH (2021), empresa mundial de pesquisas e rastreamento de consumidores de telefones, as vendas no Brasil no final de 2021 englobam 93.4 % para o sistema operacional Android e os

outros 6.6 % para as outras plataformas, incluindo o iOS, Windows Mobile, BlackBerry entre outras.

Fazer compras é uma tarefa relativamente simples, mas exige muito tempo. Na vida frenética atual da maioria da população isso é um empecilho, devido ao deslocamento, e a escolha de produtos, exigindo um tempo considerável. E para os que desejam economia esse tempo poderá dobrar facilmente.

“O consumidor está pesquisando e planejando mais antes de efetuar a compra nos supermercados” (ASSOCIAÇÃO PAULISTA DE SUPERMERCADOS, 2016, p. 2). Portanto conseguir uma economia maior requer um tempo maior, por conseguinte do planejamento e pesquisa de preços. A pesquisa de preços em vários supermercados e a compra de produtos em muitos deles, para conseguir uma economia mais efetiva, é inviável, pois o tempo de compra em vários supermercados seria exorbitante. Para conseguir esta tarefa com perfeição seria necessário utilizar uma aplicação que integrasse todos os pontos de vendas próximos usando geolocalização, e mostrar os produtos mais baratos de cada um deles.

A pandemia do COVID-19 trouxe para os consumidores novos aspectos comportamentais, com o crescimento do home office e de outros meios para seguir as restrições durante o período de pandemia. Segundo SILVA (2021) Diante do novo cenário vivenciado no mundo houve necessidade de isolamento social para evitar o contágio e propagação do vírus e com isso espaços destinados à alimentação nos restaurantes, praças de alimentação e outros foram proibidos. Conforme apontado pelo SEBRAE (2020), as opções de delivery foram mais ofertadas nesses ambientes.

Portanto, esse projeto teve como resultado o desenvolvimento de uma ferramenta para suprir essa necessidade da população e oferecer uma melhor comodidade e facilidade para os mesmos. A aplicação foi desenvolvida para dispositivos móveis utilizando o *framework* Xamarin.Forms, a título desse projeto somente para a plataforma Android.

1.1 Objetivos

1.1.1 Geral

O Objetivo geral desse trabalho foi de criar um sistema *delivery* no qual auxilia a busca pelos produtos mais baratos em lojas próximas da região. Além disso, permitir ao cliente realizar suas compras e recebê-las no endereço indicado de forma integral, diferente das entregas fracionadas utilizadas pelos aplicativos de *delivery* do mercado, normalmente uma entrega por loja.

1.1.2 Específicos

- Realizar revisão bibliográfica sobre as tecnologias que serão utilizadas no desenvolvimento.
- Realizar o levantamento de requisitos a fim de projetar corretamente a aplicação.
- Projetar a aplicação.
- Desenvolvimento da aplicação.
- Realizar a aplicação prática do sistema desenvolvido.

1.2 Relevância do trabalho

O presente trabalho apresenta a criação de um sistema de entregas para proporcionar comodidade e facilidade ao buscar produtos mais baratos na região do cliente. Com esse sistema espera-se que os clientes consigam efetuar suas compras com o menor valor possível na sua região e com a facilidade de recebê-las na sua residência. Os lojistas terão uma nova forma de vender seus produtos e possivelmente a concorrência entre os mesmos deverá gerar produtos com melhores preços. Além da criação de emprego para entregadores que irão realizar o recolhimento das compras nas lojas e entregá-las para os clientes.

1.3 Metodologia

Inicialmente, foi realizada uma revisão bibliográfica sobre o tipo de arquitetura SOA, o estilo de criação de *Web Services* REST, o *framework* Spring Boot, geolocalização de dispositivos com ênfase em aparelhos celulares e sobre desenvolvimento mobile, híbrido, multiplataforma e sobre o *framework* Xamarin. Além disso, também foi realizada uma pesquisa de trabalhos relacionados a este.

Após a revisão da literatura, foi feito o levantamento de requisitos. Foram levantados requisitos funcionais e não funcionais que devem descrever todos os aspectos significativos do aplicativo.

Após o levantamento de requisitos, o sistema foi projetado. Inicialmente foi modelado o diagrama de casos de uso, em nível de usuário e nível de sistema. Também foi realizada a modelagem de dados.

Após projetar, foi realizado o desenvolvimento dos aplicativos e a API, de forma iterativa e incremental. Inicialmente foi projetada e desenvolvida a API REST desenvolvida utilizando o *framework* Spring Boot, e em seguida foram desenvolvidos os aplicativos do cliente, entregador e lojista respectivamente.

Após o desenvolvimento, foi realizada uma aplicação prática do sistema no qual foi utilizada toda a aplicação com o propósito de verificar *bugs* nas funcionalidades, e se estavam de acordo com o que foi projetado.

1.4 Estrutura do trabalho

O restante do trabalho está estruturado como segue. No capítulo 2 são apresentadas as referências bibliográficas sobre as tecnologias utilizadas no desenvolvimento e alguns trabalhos relacionados com esse trabalho. No capítulo 3, são descritas as principais atividades realizadas durante o projeto do sistema, levantamento de requisitos funcionais e não funcionais, diagrama de casos de uso, e o projeto do banco de dados. O Capítulo 4 descreve

um estudo de caso de para apresentação das funcionalidades do sistema. E por fim, no capítulo 5 são apresentadas as conclusões do trabalho.

2 REVISÃO BIBLIOGRÁFICA

2.1 SOA

O objetivo da Arquitetura Orientada a Serviços é de integrar componentes de softwares independentes, desenvolvidos com tecnologias distintas ou não, a fim de aprimorar a eficiência, agilidade e produtividade ao implementar algo que necessite de determinado conjunto de recursos gerado pelos componentes em questão (ERL, 2009; O'BRIEN, MERSON e BASS, 2007; PAPAZOGLU e HEUVEL, 2007).

Uma implementação SOA pode constituir-se de uma junção de tecnologias, produtos, APIs, extensões da infra-estrutura de suporte entre outras partes. (ERL, 2009). A unidade fundamental da arquitetura SOA é o serviço (ERL, 2009; O'BRIEN, MERSON e BASS, 2007).

Os serviços são entidades autônomas, independentes de linguagens de programação ou sistemas operacionais específicos, e devem apoiar coletivamente uma tarefa de negócio ou processo comum. (ERL, 2009; PAPAZOGLU e HEUVEL, 2007). Eles executam funções que vão desde responder pedidos simples, à execução de processos de negócios sofisticados (PAPAZOGLU e HEUVEL, 2007).

É importante direcionar a SOA como modelo arquitetônico que seja ajustável a qualquer plataforma. No mercado atual, a plataforma de tecnologia mais associada a SOA é a de Web Services (ERL, 2009).

Conforme ERL (2009) afirma, a plataforma de Web services pode ser distribuída em duas gerações, cada uma associada a uma coleção de padrões e especificações.

A primeira geração é composta por: *Web Services Description Language* (WSDL), o *Simple Object Access Protocol* (SOAP), o *Universal Description, Discovery and Integration* (UDDI) e o *XML Schema Definition Language* (XSD) (ERL, 2009).

A segunda geração consiste em extensões WS-* que aprimoram o framework fundamental da troca de mensagens de primeira geração, esse conjunto de tecnologias *Web Services* fornece um numeroso conjunto de recursos muito mais complexo, tanto em termos da tecnologia quanto de design. (ERL, 2009). De acordo com SAUDATE (2016) as especificações das duas gerações utilizam o estilo RPC (*Remote Procedure Call*).

Existe ainda um modelo arquitetônico conhecido como REST (*Representational State Transfer*) que não adota o modelo RPC dos *Web Services* tradicionais (SAUDATE, 2016).

2.1.1 SOAP

Segundo SAUDATE (2014) ele foi criado e é mantido pela *World Wide Web Consortium* (W3C). Para MARY e VINOSHRI (2013) o SOAP é um protocolo de especificação para troca de informações. Ele utiliza na linguagem de marcação *Extensible Mark up Language* (XML) para o seu formato de mensagem, e depende de outros protocolos da camada de aplicação, principalmente *Hypertext Transfer Protocol* (HTTP) e *Simple Mail Transfer Protocol* (SMTP) para as mensagens de transação e transmissão.

De acordo com MARY e VINOSHRI (2013) o SOAP consiste em três partes, são elas: o envelope, que define o conteúdo da mensagem e como processá-la, um conjunto de regras de codificação para determinar os tipos de dados definidos da aplicação e que possam ser processados pelos nós adjacentes ao destinatário, e uma convenção de chamada de procedimentos e respostas.

SOAP possui três características principais: Extensibilidade para a adição de novos componentes, a neutralidade pois ele pode ser usado por vários protocolos de transporte (HTTP, SMTP, TCP ou JMS), e por fim, é independente de qualquer plataforma. Por utilizar vários protocolos de transporte e ser desenvolvido na linguagem de marcação XML, proporciona uma forma de comunicar aplicações em diferentes sistemas operacionais, com tecnologias e linguagens de programação distintas (MARY e VINOSHRI, 2013).

2.1.2 WSDL

De acordo com, MARY e VINOSHRI (2013) WSDL significa *Web Services Description Language*, que basicamente é um documento XML usado para descrever Web Services. Ele descreve serviços como coleções de rede ou portas. As definições abstratas de portas e as mensagens são separadas de sua utilização concreta, permitindo assim o reuso dessas definições. O WSDL também é recomendado pela W3C.

O WSDL trabalha em conjunto com o SOAP para criar e integrar Web Services, assim, quando um programa cliente habilitar um *Web Service* pode ler o arquivo WSDL para determinar quais operações estão disponíveis, além dos tipos de dados e o seu conteúdo. Após isso o cliente pode utilizar o SOAP para transportar as operações listadas no arquivo WSDL (MARY e VINOSHRI, 2013).

2.1.3 UDDI

A *Universal Description, Discovery and Integration* é uma estrutura independente de plataforma para detalhar e integrar serviços usando a internet. Utiliza um diretório para armazenar informações sobre os *Web Services* descritos pelo modelo WSDL e comunica-se através do SOAP. O UDDI foi desenvolvido na plataforma Microsoft .NET (MARY e VINOSHRI, 2013).

Segundo MARY e VINOSHRI (2013) antes do UDDI, não havia um meio de integrar sistemas e processos, muito menos, de expor informações sobre produtos e serviços, das empresas para seus clientes e parceiros, ou até mesmo, efetuar vendas online, algo muito comum atualmente.

De acordo com MARY e VINOSHRI (2013) um provedor de serviços web adota o UDDI para publicar o serviço e utiliza o padrão WSDL para a descrição dos dados, e o SOAP, para transportar esses dados. Assim, uma empresa pode criar um padrão UDDI para um determinado tipo negócio e as empresas deste ramo de negócio poderiam registrar seus produtos ou serviços em um diretório UDDI utilizando o padrão criado anteriormente, desse modo, uma aplicação poderia ser criada com o objetivo de acessar o diretório e passar as informações dos produtos das empresas para os respectivos clientes interessados.

2.1.4 REST

O REST (*Representational State Transfer*) é um estilo de criação de *Web Services* e foi elaborado por Roy Fielding, coautor do protocolo HTTP. O REST utiliza boas práticas de uso do HTTP como: Uso apropriado dos métodos GET, PUT, POST e DELETE, uso do código de status padronizados para representar falhas ou sucessos, uso adequado de cabeçalhos HTTP, interligar vários recursos diferentes e uso apropriado de URLs (*Universal Resource Locator*) (KENNEDY *et al.*, 2011; SAUDATE, 2016).

Conforme MORO, DORNELES E REBONATTO (2011) os *Web Services* no estilo REST podem trabalhar nas representações em XML ou JSON (JavaScript Object Notation) com flexibilidade de trabalhar tanto com WSDL quanto WADL (*Web Application Description Language*).

Segundo FIELDING (2000), conforme citado por MORO, DORNELES E REBONATTO (2011) é possível desenvolver um sistema que segue as restrições impostas pelo estilo arquitetural REST sem utilizar HTTP e sem interagir com a Web.

De acordo com KENNEDY *et al.* (2011) os métodos do protocolo HTTP são utilizados para realizar as seguintes tarefas:

- GET: Recupera a representação de determinado recurso.
- PUT: Se o recurso identificado pela URI (*Universal Resource Identifier*) não existir, então o recurso é criado, caso contrário o recurso é atualizado.
- POST: Cria sub-recursos dentro dos recursos existentes. Por exemplo, um recurso /produtos encontra-se no servidor, e há necessidade de criar um novo produto chamado arroz. Desse modo pode-se utilizar um POST para criar esse novo recurso com a URI /produtos/arroz. A diferença entre o PUT e o POST é quem determina a URI. No PUT o servidor define a URI, já no POST o cliente determina a URI.
- DELETE: Exclui um recurso existente.

Para SAUDATE (2016) em REST, tudo é definido em termos de recursos, sendo estes, os dados que trafegam pelo protocolo. A representação dos recursos é realizada através de URIs.

A utilização do paradigma arquitetônico REST pode ser usado de diversas maneiras, RESTful é uma delas. De acordo com LI (2011) RESTful é um tipo de *Web Services* que utiliza a arquitetura REST. Já conforme MORO, DORNELES E REBONATTO (2011) sistemas que seguem os princípios do estilo arquitetural REST podem ser referenciados também como “RESTful”.

O RESTful tem desempenho superior, se comparado aos *Web Services* baseados em SOAP. Segundo SOUZA *et al.* (2012) a abordagem RESTful é superior na utilização do método GET, desse modo, aplicações distribuídas que trafegam muitas informações em suas mensagens podem ter um ganho de desempenho. Em projetos nos quais a troca de informação não é grande pode-se considerar a adoção de *Web Services* com SOAP para desfrutar das extensões WS-*

2.1.4.1 Spring Boot

De acordo com BOAGLIO (2017) A ideia inicial veio da necessidade do Spring Framework ter suporte a servidores web embutidos.

A equipe do Spring percebeu que existiam outras pendências, como fazer aplicações prontas para nuvem (*cloud-ready applications*). Depois, com a anotação `@Conditional` criada no Spring Framework 4, possibilitou que o Spring Boot fosse criado oficialmente (BOAGLIO, 2017).

Para BOAGLIO (2017) é uma maneira eficiente e eficaz de criar aplicações Spring facilmente e colocá-las no ar, funcionando independente de servidores de aplicação. O Spring Boot criou um conceito novo até o momento inexistente na especificação JEE (*Java Enterprise Edition*), que acelera o desenvolvimento e simplifica bastante a vida de quem trabalha com aplicações do Spring Framework, mas não altera em nada para quem desenvolve com as especificações oficiais (com EJB, CDI ou JSF).

Ainda segundo BOAGLIO (2017) ele pode ser confundido com um simples *framework*, mas na verdade ele é um conceito totalmente novo para criar aplicações web. No conceito de Java Web Container, temos o framework Spring, controlando as regras de negócio empacotadas em um JAR, e ele deverá obedecer os padrões do servidor de aplicação. Na figura 1 é demonstrado como o Spring Framework funciona.

Figura 1 – Spring Framework com Java Web Container (BOAGLIO, 2017)



No conceito novo, temos o Spring Boot no controle total, providenciando o servidor web e controlando as suas regras de negócio, demonstrado na figura 2 (BOAGLIO, 2017).

Figura 2 – Spring Boot no controle total da aplicação web (BOAGLIO, 2017)



2.2 Geolocalização

A geolocalização pode identificar a posição de uma pessoa ou objeto na terra. (DJUKNIC E RICHTON, 2001; MONTEIRO, 2015). A posição geográfica de uma entidade pode ser definida pelos componentes de latitude e longitude, e podem ser obtidas por dispositivos que utilizam alguma tecnologia de geolocalização (MONTEIRO, 2015).

Segundo ROXIN *et al.*, (2007) as tecnologias de geolocalização podem ser baseadas em rede ou internet como o CID (Identificação de Célula), RSS (Força do sinal recebida), Multipath/Fingerprint, AOA (*Angle Of Arrival*), TDOA (*Time Difference Of Arrival*), TOA (*Time Of Arrival*), OTDOA *Observed Time Differnce Of Arrival*, U-TDOA (*Uplink Time Difference of Arrival*) ou nos métodos com emprego de satélites, muito utilizados no desenvolvimento de novos aplicativos para dispositivos móveis, principalmente por sua alta precisão e facilidade de acesso em smartphones e tablets atuais, são eles: o GPS (Sistema de Posicionamento Global) ou o A-GPS (Sistema de Posicionamento Global Assistido).

2.2.1 GPS

Uma funcionalidade importante dos *smartphones* é a de geolocalização, muitas aplicações utilizam a localização para determinados fins, principalmente para mostrar locais próximos referentes aos objetivos do usuário. Uma das formas de determinar um ponto na superfície da terra é o GPS (*Global Positioning System*). O Sistema de Posicionamento Global

é considerado um dos métodos de geolocalização mais precisos, ele utiliza satélites para determinar a posição de uma pessoa, objeto ou lugar. A entidade precisa de um hardware e software para receber o sinal GPS (ROXIN *et al.*, 2007).

De acordo com DJUKNIC e RICHTON (2001) o GPS consistia inicialmente em uma constelação de 24 satélites, igualmente espaçados em seis planos orbitais de 20.200 quilômetros acima da Terra, que transmitem dois tipos de sinais codificados, o L1 para uso civil e o L2 para forças armadas e uso do governo.

O Sistema de Posicionamento Global foi criado pelo departamento de defesa dos Estados Unidos da América na década de 1970 para uso militar. (CORAZZA, 2014). O sinal L1 com uma taxa de acerto de 100 metros para questões de segurança e a frequência L2 com precisão de até 10 metros. Em maio de 2000 o sinal ficou totalmente livre para o uso civil, a justificativa do governo foi de que o GPS iria atender a demanda civil e comercial. (DJUKNIC e RICHTON, 2001; OLIVEIRA, 2013). Ainda de acordo com OLIVEIRA (2013) a constelação atualmente é de 30 satélites.

Segundo ROXIN *et al.*, (2007), os satélites são equipados com relógios que mantêm um tempo de precisão de até 3 ns (nanosegundos). A precisão é de primordial importância para determinar exatamente quanto tempo leva para os sinais viajarem a partir de cada satélite GPS e determinar a posição do objeto.

Os satélites emitem sinais de rádio que são captados pelos aparelhos receptores GPS, estes aparelhos usam sinais de no mínimo 4 satélites e calculam a distância entre a recepção de cada um deles, pelo intervalo de tempo entre o momento atual e o momento em que os sinais foram enviados, com essa pseudodistância calculada é possível efetuar a triangulação das posições e identificar a coordenada geográfica em que o objeto se encontra (CANALLE, 2011).

De acordo com ROXIN *et al.*, (2007) o GPS possui um atraso relativamente longo para obter uma posição inicial do objeto. Para DJUKNIC e RICHTON (2001) o GPS também precisa de um clima aberto para o sinal transitar livremente.

2.2.2 A-GPS

Segundo HOENTSCH (2013) o A-GPS (Sistema de posicionamento global assistido) é uma versão melhorada do GPS que recebe informações através de uma conexão de dados como 3G ou GPRS.

O GPS assistido oferece uma melhor precisão se comparado ao GPS (DJUKNIC e RICHTON, 2001; ROXIN *et al.*, 2007) além de resolver o problema do longo atraso ao determinar a localização da entidade (DJUKNIC e RICHTON, 2001).

De acordo com HOENTSCH (2013) a rapidez se dá porque a conexão inicial não é feita diretamente com o satélite, como funciona com o GPS convencional, mas sim com uma antena de telefonia celular que possui a localização destes satélites armazenadas e as transmite para o GPS com uma velocidade até 40 vezes maior.

O A-GPS só é disponível para dispositivos com conexão GPRS (*General Packet Radio Service*). O GPRS é um serviço que permite o envio e recepção de informações através de uma rede telefônica móvel, este recurso é mais comum em celulares com função GPS do que em dispositivos específicos de GPS (HOENTSCH, 2013).

2.3 Desenvolvimento Mobile

Segundo IBGE (2020), conforme citado por PAIVA (2020) Pesquisas indicam que só no Brasil em 2020, cerca de 79,3% dos brasileiros possuíam um *smartphone*, e com base em dados dos anos anteriores, esse número continuará a crescer.

Diversas utilidades são implementadas por meio de aplicações para atender a grande quantidade de usuários, que são desde funcionalidades simples como despertador ou agenda, até mais complexas como envio de *e-mails*, jogos e reprodução de músicas e filmes. De acordo com LOPES (2013) as aplicações são softwares desenvolvidos com a finalidade de serem executados em plataformas móveis, como *tablets e smartphones*.

De maneira geral, os aplicativos são desenvolvidos para sua linguagem nativa e para o seu sistema operacional específico como iOS com Swift e Android com Java ou Kotlin, e caso seja de interesse disponibilizá-los para mais de uma plataforma torna necessário desenvolvê-los mais de uma vez com tecnologias diferentes. Com isso, o custo para desenvolver uma aplicação móvel torna-se muito caro por demandar infraestrutura apropriada, maior conhecimento e licenças para desenvolvimento como destaca BOCARD (2021).

Como resultado disso, surgiram várias abordagens de desenvolvimento que trazem como principal proposta a construção de um único aplicativo que possa ser disponibilizado para grande parte das plataformas do mercado (SILVA; SANTOS, 2014). Podemos destacar dois tipos, as aplicações híbridas e desenvolvimento multiplataforma.

2.3.1 Aplicações Híbridas

Segundo JOBE (2013), as aplicações híbridas são uma mescla de aplicações nativas e web e geralmente são desenvolvidas utilizando Javascript, HTML e CSS. Esse tipo de aplicação possui um alcance maior, pois não é necessário o download da aplicação.

Contudo, de acordo com ROMMI (2021) essas aplicações não conseguem utilizar recursos nativos dos dispositivos em que são executadas e isso é um fator limitante para suas possibilidades de uso. Além disso, sempre necessitam de conexão com a internet para possibilitar seu acesso.

2.3.2 Desenvolvimento Multiplataforma

É a prática de desenvolver aplicações para múltiplas plataformas ou ambientes de software. Desenvolvedores e engenheiros utilizam vários métodos para acomodar diferentes sistemas operacionais ou ambientes para uma aplicação ou produto (TECHOPEDIA, 2020).

Normalmente o desenvolvimento multiplataforma é mais fácil, pois utiliza linguagens comuns assim evitando necessidade de especialização nas linguagens específicas de cada sistema operacional. Também, a utilização de ferramentas multiplataformas torna desnecessário o estudo detalhado de cada plataforma como afirma CONSTANTIN *et al.*, (2016). Ainda segundo CONSTANTIN *et al.*, (2016) as ferramentas multiplataformas

utilizam somente suas APIs e seus próprios recursos e a maior vantagem do desenvolvimento multiplataforma é a facilidade que o software possui em atingir um grande público em um curto período de tempo.

Várias empresas criaram *frameworks* para desenvolvimento de aplicações multiplataformas, a seguir será descrito o *framework* utilizado nesse trabalho.

2.3.2.1 Xamarin

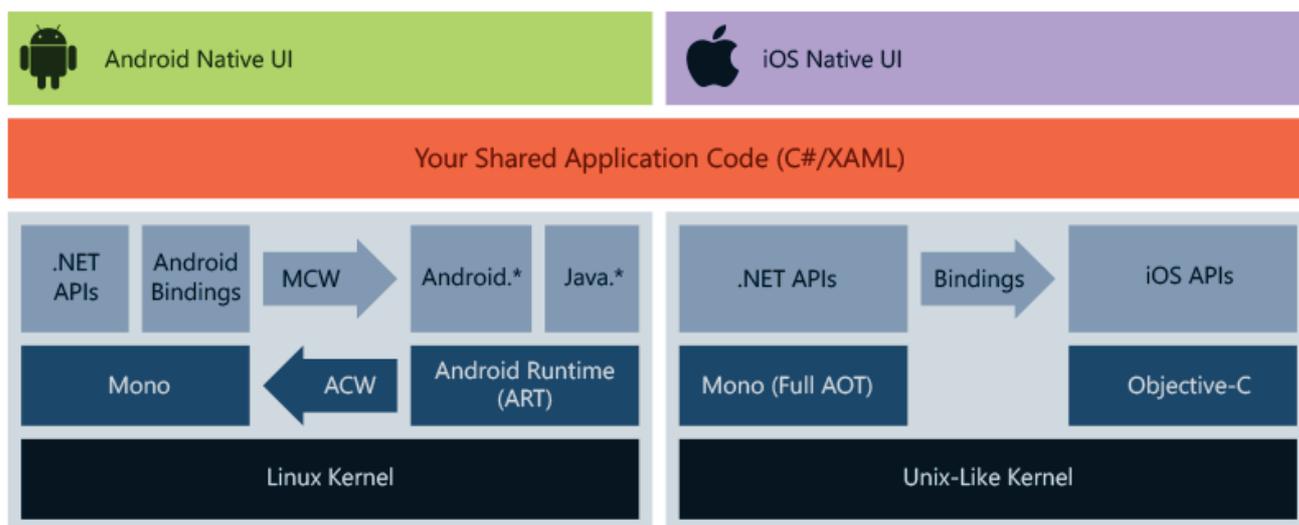
Empresa fundada em 2011, inicialmente desenvolveu a tecnologia Mono, que é a implementação *open source* do *framework* da Microsoft .NET para C#. Com essa implementação outras plataformas podem utilizar essa tecnologia (TANNER 2016).

Xamarin é uma plataforma de software livre para a criação de aplicações modernas e de alto desempenho para as plataformas iOS, Android e Windows com o .NET. O Xamarin é uma camada de abstração que gerencia a comunicação de código compartilhado com o código da plataforma subjacente (XAMARIN, 2021).

O Xamarin permite os desenvolvedores compartilharem em média 90% do seu aplicativo entre as plataformas. Esse padrão permite que durante o desenvolvimento seja gravado toda a lógica de negócios em uma única linguagem e alcancem o desempenho e a aparência nativos em cada plataforma (XAMARIN, 2021).

A figura 3 mostra a arquitetura geral de um aplicativo Xamarin multiplataforma. O *framework* permite a criação de uma interface do usuário nativa em cada plataforma e a escrita da lógica de negócios em C# que é compartilhada entre as plataformas. Na maioria dos casos, cerca de 80% dos códigos do aplicativo são compartilháveis ao utilizar o Xamarin. (XAMARIN, 2021).

Figura 3 – Arquitetura do Xamarin (XAMARIN, 2021)



O Xamarin.Forms é uma melhoria no *framework* que permite aumentar ainda mais o compartilhamento já que desenvolve um mesmo código de interface gráfica para Android, iOS, tvOS, watchOS, macOS e Windows. O Desenvolvimento da interface pode ser utilizando C# ou sua própria linguagem de marcação, XAML (*Extensible Application Markup Language*) (XAMARIN, 2021).

2.4 Trabalhos Relacionados

Nesta seção serão apresentados alguns aplicativos com finalidades semelhantes ao trabalho em questão. As propostas representam trabalhos moderadamente e fortemente relacionados.

2.4.1 FASTLINE: Aplicativo mobile para lanchonetes

Melo *et al.* (2015) desenvolveram um sistema nomeado de FastLine e tem como objetivo melhorar o processo de venda das lanchonetes escolares durante o intervalo das aulas. De acordo com MELO *et al.*, (2015) o intervalo muitas vezes é curto em função do número de alunos que compram no estabelecimento. O sistema é composto pelas seguintes plataformas: um aplicativo mobile para o cliente realizar seus pedidos e uma aplicação *web* para a lanchonete receber os pedidos. As plataformas são integradas por meio de um *webservice* utilizando SOA. Por meio do módulo da aplicação Web, utilizado na lanchonete, é possível coletar os pedidos tanto dos clientes que realizam os pedidos presencialmente quanto dos clientes que realizarão seus pedidos por meio da aplicação mobile. A aplicação mobile permite que o cliente escolha seus produtos e agende o horário da retirada do pedido. O

cliente é informado em quanto tempo seu pedido estará pronto baseado na fila atual de pedidos, de acordo com um cálculo da estimativa de tempo de preparação de cada produto, ou se desejar, o cliente pode agendar um horário disponível pelo aplicativo para retirar seu pedido.

2.4.2 Uber Eats

O Uber Eats é uma plataforma de entrega de alimentos da empresa Uber Technologies Inc. Que conta com o aplicativo para *smartphone*. Está presente em mais de vinte países do mundo, inclusive no Brasil, ajudando mais de 600.000 restaurantes a se conectarem com entregadores parceiros para atender mais pedidos (UberEats, 2021).

Segundo Uber conforme citado por Nigri (2019) O UberEats é uma plataforma de intermediação de entregas que leva seus pratos favoritos até você de forma simples e eficaz, através dos entregadores parceiros cadastrados. No app, você pode escolher entre os menus completos de alguns dos melhores restaurantes da sua região e pedir seus pratos favoritos quando desejar.

2.4.3 Extra Delivery

O Extra *Delivery* é um sistema no qual o usuário pode fazer compras de produtos alimentícios de todos os tipos. O cliente pode escolher todos os produtos desejados e solicitar a entrega. Os produtos são entregues por lojas físicas da rede de supermercados Extra de acordo com a localização do usuário, mas, a cobertura de entregas é apenas nas regiões metropolitanas de São Paulo, Rio de Janeiro e Distrito Federal (EXTRA, 2021).

2.4.4 Quero Delivery

Fundado em 2018, o Quero Delivery é uma empresa sergipana que desenvolveu seu aplicativo de delivery homônimo e aposta em cidades menores. Enquanto a maioria das plataformas de delivery apostavam nas capitais, o quero delivery focou no caminho inverso, nas cidades do interior, o que deu muito certo chegando ao crescimento de 400% com o impulso da pandemia (MOVIMENTO ECONÔMICO, 2021).

O app está disponível para android e iOS. E ele trabalha em duas frentes. Inicialmente, era um app de entregas das quais cada cidade que possui as lojas parceiras, e onde os clientes da mesma cidade podem realizar seus pedidos, fazia intermediação do negócio da loja parceira por meio da plataforma. Atualmente, também opera em sistema de *full service*, que é quando o cliente também usa a plataforma para concentrar os pedidos e a entrega também pelo conta do Quero Delivery (REDE FOOD SERVICE, 2021).

2.4.5 iFood

Os desenvolvedores criaram um sistema delivery de restaurantes para os dispositivos que utilizam os Sistemas operacionais, Android, iOS e Windows Mobile, além da aplicação Web. No sistema denominado iFood é possível acessar a localização do cliente pelo CEP ou GPS, assim é exibido os restaurantes próximos que possuem cadastro no sistema. O cliente pode escolher seu pedido e a forma de pagamento. Não é possível pagar pelo aplicativo, o pagamento é efetuado na entrega. No restaurante o usuário recebe os pedidos pela conta do restaurante através de notificações e prepara-os (IFOOD, 2016). O sistema integra várias aplicações de diferentes tecnologias e tem como principal objetivo facilitar a comunicação dos clientes e restaurantes.

2.4.6 Bom Delivery: Um aplicativo de delivery genérico para cidades do interior

Melo (2019) desenvolveu um aplicativo que permite a intermediação da venda de produtos de estabelecimentos delivery de forma intuitiva. O Aplicativo traz algumas vantagens tanto ao dono do estabelecimento que tem uma plataforma online para listagem e venda de seus produtos, quanto para o usuário que tem de forma simples o acesso a vários estabelecimentos e produtos através de seu *smartphone*. Além disso, foi desenvolvido outro aplicativo auxiliar para o dono de estabelecimento receber e aceitar os seus pedidos. O dono do estabelecimento não é responsável por cadastrar o estabelecimento e produtos, isso deve ser feito automaticamente através de especificações escritas, ao fim do cadastro o lojista receberá o login e senha para efetuar a autenticação e receber os pedidos. Já o cliente terá acesso aos estabelecimentos disponíveis para compra de produtos selecionados por ele.

3 PROJETO BUSCOU BARATO

Neste capítulo será apresentado o projeto do sistema, ele é composto por três módulos, cada módulo possui uma aplicação mobile para cada tipo de usuário. Para fins desse projeto, apenas as versões da plataforma Android serão demonstradas. Além destes, uma API para comunicação entre as aplicações também foi desenvolvida.

As aplicações de cliente, entregador e lojista foram desenvolvidas na linguagem C# e juntamente com o *framework* Xamarin.Forms, para desenvolvimento de aplicações móveis nativas, que podem ser disponibilizadas para Android, iOS, tvOS, watchOS, macOS e Windows. As aplicações comunicam-se com uma API desenvolvida na linguagem Java e o *framework* Spring Boot. Além do banco de dados PostgreSQL.

Serão descritos nos tópicos a seguir, os diagramas de casos de uso, requisitos funcionais e não funcionais para cada módulo do sistema e o projeto do banco de dados.

3.1 Buscou Barato Módulo Cliente

A aplicação do cliente é utilizada principalmente para realização das compras, na qual o cliente pode adicionar e remover produtos, além de escolher o endereço de entrega e verificar o andamento das entregas até chegarem ao endereço indicado.

3.1.1 Requisitos funcionais e não funcionais

Os requisitos funcionais, estão resumidos no Quadro 1, são eles: 1) Cadastrar Usuário Cliente. Para ter acesso as funcionalidades do aplicativo; 2) Autenticar Usuário. É necessária para efetuar compras no aplicativo, e gerenciar as informações da conta como compras já realizadas, configuração de senha e endereço. Para verificar os preços das lojas próximas não é necessário autenticar-se no sistema pois o aplicativo por padrão coleta a localização do cliente e busca as lojas próximas. 3) Gerenciar produtos do carrinho. Esse requisito permite adicionar, alterar, remover e consultar produtos do carrinho de compras. 4) Gerenciar compras. Esse requisito inclui finalizar, detalhar, recriar, listar compras já realizadas e rastrear entrega. 5) Alterar senha. Esse requisito permite alterar a senha do usuário. 6) Alterar endereço de entrega. Esse requisito permite alterar o endereço, anteriormente cadastrado no

momento do cadastro do usuário para realização da entrega das compras do cliente. 7) Pesquisar Produtos. O cliente pode utilizar a pesquisa de produtos e verificar os produtos desejados das lojas próximas.

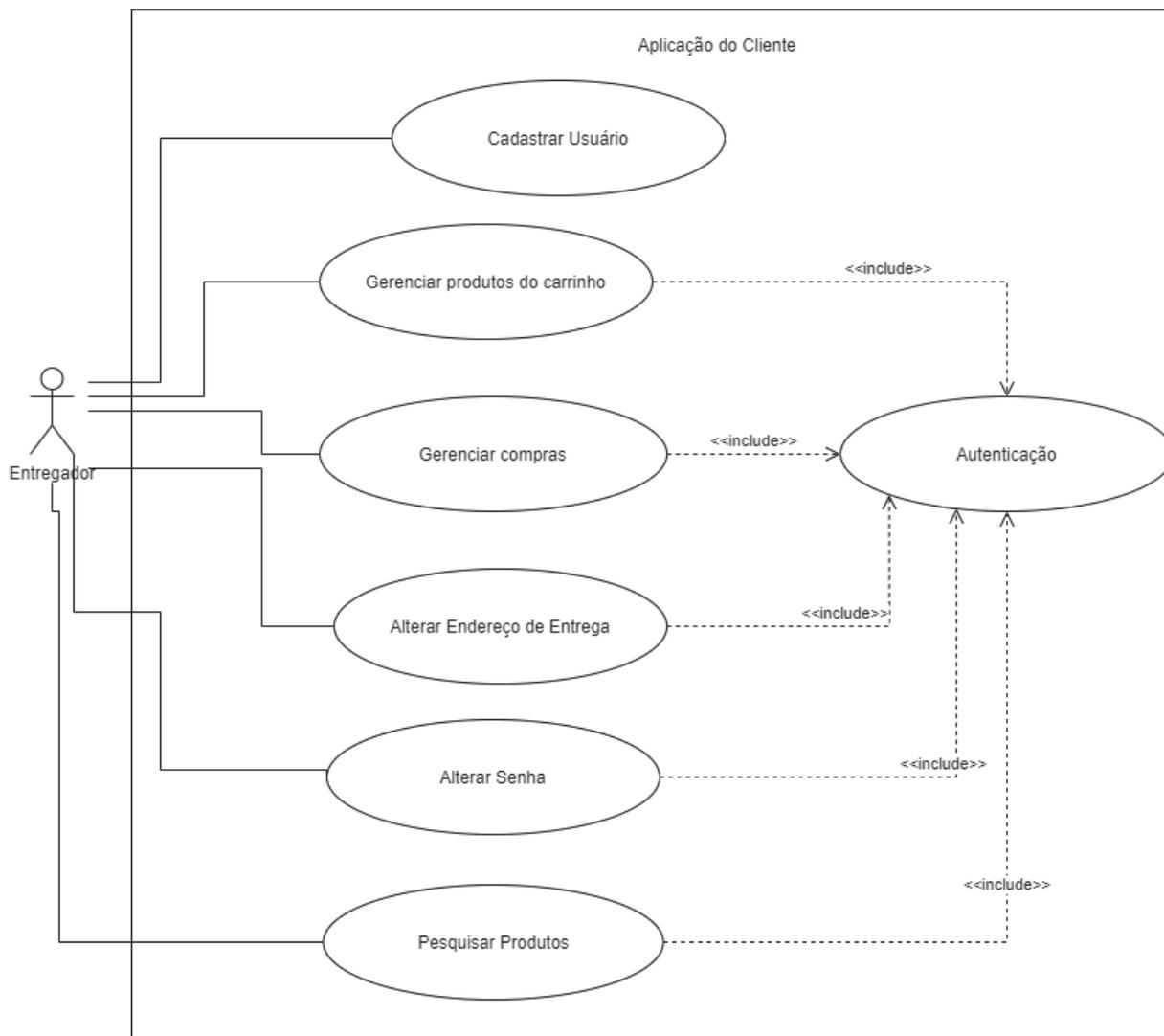
Quadro 1 - Requisitos funcionais e não funcionais da ferramenta.

Requisitos Funcionais
R01: Cadastrar Usuário Cliente
R02: Autenticar Usuário
R03: Gerenciar produtos do carrinho (Adicionar, Alterar, Remover e Consultar)
R04: Gerenciar compras (Finalizar, Detalhar e Recriar)
R05: Alterar Senha
R06: Alterar Endereço de Entrega
R07: Pesquisar Produtos
Requisitos Não-Funcionais
R08: O cliente só poderá acessar as funcionalidades do aplicativo se autenticado
R11: O sistema estará disponível para as plataformas, Android, iOS e UWP
R12: O sistema deverá ser desenvolvido com a linguagem C# utilizando o Xamarin.Forms
R13: A API deverá ser desenvolvida com a linguagem Java utilizando o SpringBoot
R14: A API irá utilizar o JPA Hibernate para comunicação com o banco de dados
R15: A API deverá utilizar o banco de dados Postgresql
R16: O sistemas precisam de acesso a internet para funcionar

3.1.2 Casos de Uso

Em conformidade com os requisitos já citados, foram estabelecidos os seguintes casos de uso: Cadastrar usuário; Autenticação; Gerenciar produtos do carrinho; Gerenciar compras; Alterar senha; Alterar endereço de entrega; Pesquisar produtos. Abaixo é demonstrado o diagrama de casos de uso.

Figura 4 - Diagrama de casos de uso da aplicação do cliente



3.1.2.1 Casos de Uso - Nível de Usuário

Na análise inicial foram identificados os requisitos essenciais para a aplicação, o diagrama de casos de uso foi modelado e está representado na figura 4. Assim, é possível verificar algumas funcionalidades do módulo do usuário.

O caso de uso Autenticação verifica os dados informados, login e senha, para permitir o acesso as funcionalidades da aplicação. Sem realizar a autenticação é possível somente verificar produtos da lista de lojas próximas. As funcionalidades principais como adicionar produtos no carrinho de compras, visualizar compras já realizadas, configurar endereço e senha, necessitam que o usuário esteja logado para funcionar. O requisito R02 é associado.

O caso de uso Cadastrar usuário é responsável por receber e salvar os dados do usuário para que seja utilizado a autenticação no futuro e utilizar as funcionalidades da aplicação. Associa-se ao requisito R01.

No caso de uso Gerenciar produtos do carrinho é possível adicionar, editar, remover e listar os produtos da compra ainda não finalizada. Também é possível verificar a quantidade de itens e o valor total desses produtos antes da finalização da compra. Está relacionado ao requisito R03.

O caso de uso Gerenciar Compras permite o usuário verificar o histórico de compras realizadas por ele, visualizar os detalhes da compra, a lista de produtos e o valor total. Também é possível recriar a compra caso esta já esteja com o Status Entregue que adiciona todos os produtos da antiga compra no carrinho. Caso a compra esteja no processo de coleta é possível rastrear a última parada do entregador e verificar sua localização. Possui o requisito R04 associado.

No caso de uso Alterar Senha é possível mudar a senha do usuário, informando a senha atual e a nova senha. A senha deve conter no mínimo 8 caracteres, pelo menos uma letra maiúscula, pelo menos uma letra minúscula, pelo menos um número e pelo menos um caractere especial. Caso não esteja no padrão citado é mostrado uma mensagem informando que a senha precisa estar no padrão do sistema. O requisito R05 está associado a esse caso de uso.

O caso de uso Alterar endereço de entrega tem como objetivo realizar a edição do local de entrega das compras, anteriormente informada durante o cadastro do usuário. É alertado que o carrinho de compras será limpo antes da finalização da edição, para não haver produtos de regiões diferentes da localização do endereço escolhido neste momento. Tem o requisito R06 associado.

O caso de uso Pesquisar produtos permite que o usuário consiga buscar produtos pelo nome e que estão em estoque nas lojas da sua região. A lista de produtos disponível no app é filtrada através do endereço informado pelo cliente, cada cidade possui um raio pré-

determinado que é utilizado para comparar a distância da loja e o endereço do cliente, caso a loja esteja dentro do raio determinado os produtos dela são mostrados para esse usuário no app e podem ser pesquisados por ele. Esse caso de uso está associado ao requisito R07.

3.1.2.2 Casos de Uso - Nível de Sistema

Serão apresentados nesta seção os casos de uso a nível de sistema. Nestes serão demonstrados o objetivo de cada função, os usuários indicados, além da pré-condição e pós-condição para realização do caso de uso. Os fluxos principais, alternativos e de exceção também serão detalhados para compreensão das interações de cada funcionalidade do sistema.

3.1.2.2.1 Caso de Uso Cadastrar Usuário

Quadro 2: Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.

Objetivo	Realizar o cadastro para acessar as funcionalidades da aplicação
Ator	Cliente
Pós-Condição	Usuário poderá acessar as funcionalidades do módulo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção Login.	4. Sistema exibe tela de login.
5. Ator toca no Botão Cadastre-se Grátis.	6. Sistema exibe a tela de cadastro.
7. Ator informa os dados cadastrais e clica em Criar Conta.	8. Sistema salva os dados e redireciona para a tela inicial.
Fluxo Exceção	
Linha 8: Dados não válidos. Sistema exibe mensagem “Algumas informações não foram preenchidas, é necessário informar todas os campos para finalizar o cadastro”, retornando ao passo 7	

3.1.2.2.2 Caso de Uso Autenticação

Quadro 3: Descrição do Caso de Uso em Nível de Sistema - Autenticação.

Objetivo	Efetuar a autenticação para acessar as funcionalidades da aplicação
Ator	Cliente
Pós-Condição	Usuário deve ser autorizado acessar as funcionalidades do módulo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator informa usuário e senha e clica em Entrar.	4. Sistema verifica se os dados são válidos.
	5. Sistema permite o acesso das funcionalidades da aplicação.
Fluxo Exceção	
Linha 4: usuário ou senha informados incorretamente. Sistema exibe mensagem “Ocorreu um erro ao logar, por favor tente novamente”, retornando ao passo 3.	
Linha 4: usuário não preencheu os campos de login ou senha. Sistema exibe mensagem “Para entrar é necessário informar o campo de usuário e senha!”, retornando ao passo 3.	

3.1.2.2.3 Caso de Uso Gerenciar Produtos do Carrinho

Seção: Adicionar Produto ao carrinho

Quadro 4: Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Adicionar produto ao carrinho.

Objetivo	Adicionar produto no carrinho de compras
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O Produto estará no carrinho de compras do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no produto que deseja adicionar.	4. Sistema exibe tela com as informações do produto.

5. Ator informa a quantidade do produto que deseja e toca no botão do carrinho.	6. Sistema salva os dados e redireciona para tela inicial.
Fluxo Alternativo	
Linha 3: Ator toca no botão do carrinho no produto que deseja adicionar.	
Linha 5: Ator toca no botão de fechar. Retorna a tela anterior.	
Fluxo Exceção	
Linha 6: Erro ao tentar cadastrar. Sistema exibe mensagem “Ocorreu um erro ao tentar adicionar o produto ao carrinho”, retornando ao passo 5.	

Seção: Remover Produto do carrinho

Quadro 5: Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Remover produto ao carrinho.

Objetivo	Remover produtos do carrinho de compras
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O produto será removido do carrinho de compras do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no botão do carrinho de compras.	4. Sistema exibe tela com os produtos que estão no carrinho e o valor total dos produtos.
5. Ator toca no Botão de Remover do Item que deseja excluir do carrinho de compras	6. Sistema exibe a tela com a mensagem “Realmente deseja remover esse produto?” e com os botões Sim e Não.
7. Ator toca no botão Sim.	8. Sistema remove o produto e retorna para a tela do carrinho de compras.
Fluxo Alternativo	
Linha 7: Ator toca no botão Não. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Erro buscar produtos do carrinho de compras. Sistema exibe mensagem “Não foi buscar os produtos do carrinho de compras”, retornando ao passo 2.	
Linha 8: Erro ao remover produto. Sistema exibe mensagem “Não foi possível remover o produto do carrinho”, retornando ao passo 4.	

Seção: Alterar Produto do carrinho

Quadro 6: Descrição do Caso de Uso em Nível de Sistema - Gerenciar produtos do carrinho - Alterar produto ao carrinho.

Objetivo	Alterar produtos do carrinho de compras
-----------------	---

Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O produto será alterado no carrinho de compras do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no botão do carrinho de compras.	4. Sistema exibe tela com os produtos que estão no carrinho e o valor total dos produtos.
5. Ator toca no Botão de Editar do Item que deseja alterar do carrinho de compras.	6. Sistema exibe a tela com as informações do produto.
7. Ator muda a quantidade e toca no botão Editar.	8. Sistema retorna para a tela do carrinho de compras com o produto alterado.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Erro buscar produtos do carrinho de compras. Sistema exibe mensagem “Não foi buscar os produtos do carrinho de compras”, retornando ao passo 2.	
Linha 8: Erro ao remover produto. Sistema exibe mensagem “Não foi possível editar o produto do carrinho”, retornando ao passo 4.	

3.1.2.2.4 Caso de Uso Gerenciar Compras

Seção: Finalizar Compra

Quadro 7: Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Finalizar Compra.

Objetivo	Finalizar compra para dar início a coleta dos produtos
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A compra será finalizada e seu status será modificado para Aguardando Início Coleta
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no botão do carrinho de compras.	4. Sistema exibe tela com os produtos que estão no carrinho e o valor total dos produtos.

5. Ator toca no Botão Finalizar Compra.	6. Sistema exibe a mensagem “Sucesso. Compra finalizada com sucesso! A coleta será realizada e será entregue na sua casa!”.
7. Ator toca no botão OK.	8. Sistema retorna para a tela inicial.
Fluxo Exceção	
Linha 4: Erro buscar produtos do carrinho de compras. Sistema exibe mensagem “Não foi buscar os produtos do carrinho de compras”, retornando ao passo 2.	
Linha 8: Erro ao finalizar a compra. Sistema exibe mensagem “Ocorreu um erro ao finalizar a compra, por favor tente novamente”, retornando ao passo 5.	

Seção: Detalhes da Compra

Quadro 8: Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Detalhes Compra.

Objetivo	Verificar detalhes de compra já finalizada
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	Verificar lista de produtos e valor total de compra já finalizada
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Compras.	4. Sistema exibe tela lista de compras já finalizadas pelo cliente.
5. Ator toca na compra que deseja verificar os detalhes.	6. Sistema exibe uma tela com as opções que é possível fazer com uma compra já finalizada.
7. Ator toca no botão Detalhes da Compra.	8. Sistema exibe tela com os produtos que foram comprados e o valor total da compra.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 8: Erro ao buscar informações da compra. Sistema exibe mensagem “Ocorreu um erro ao buscar informações da compra”, retornando ao passo 5.	

Seção: Recriar Compra

Quadro 9: Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Recriar Compra.

Objetivo	Recriar compra já finalizada
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema

Pós-Condição	Os produtos da compra anterior são adicionados no carrinho de compras
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Compras.	4. Sistema exibe tela lista de compras já finalizadas pelo cliente.
5. Ator toca na compra que deseja verificar os detalhes.	6. Sistema exibe uma tela com as opções que é possível fazer com uma compra já finalizada.
7. Ator toca no botão Detalhes da Compra.	8. Sistema exibe tela com os produtos que foram comprados e o valor total da compra.
9. Ator toca no botão Recriar Compra.	10. Sistema adiciona todos os produtos no carrinho de compras e mostra a mensagem “Sucesso. Os produtos da compra foram adicionados ao carrinho.”.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	
Linha 7: Ator toca no botão Recriar compra. Vai para a linha 10 .	
Fluxo Exceção	
Linha 8: Erro ao buscar informações da compra. Sistema exibe mensagem “Ocorreu um erro ao buscar informações da compra”, retornando ao passo 5.	
Linha 10: Erro ao recriar compra. Sistema exibe mensagem “Ocorreu um erro ao adicionar os produtos ao carrinho!”, retornando ao passo 9.	

Seção: Rastrear Entrega

Quadro 10: Descrição do Caso de Uso em Nível de Sistema - Gerenciar compras - Rastrear Entrega.

Objetivo	Rastrear compra já finalizada com o status Em Coleta
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	Mostra no Google Maps a distância do entregador até o local de entrega
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Compras.	4. Sistema exibe tela lista de compras já finalizadas pelo cliente.

5. Ator toca na compra que deseja verificar os detalhes.	6. Sistema exibe uma tela com as opções que é possível fazer com uma compra já finalizada.
7. Ator toca no botão Rastrear Entrega.	8. Sistema abre o Google Maps e mostra a última localização do Entregador.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	

3.1.2.2.5 Caso de Uso Alterar Senha

Quadro 11: Descrição do Caso de Uso em Nível de Sistema - Alterar Senha.

Objetivo	Mudar senha cadastrada anteriormente
Ator	Cliente
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A senha para um novo login é modificada
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Configurações.	4. Sistema exibe tela com os campos para informar a senha atual, a nova senha e a confirmação da nova senha.
5. Ator informa a senha atual, a nova senha e a confirmação da senha e toca no botão Trocar Senha.	6. Sistema exibe a mensagem “Sucesso. A senha atual foi atualizada com sucesso” e retorna para a tela inicial.
Fluxo Exceção	
Linha 6: Ator informa a senha atual incorretamente. O sistema exibe a mensagem “A senha atual está incorreta”.	
Linha 6: Ator informa a nova senha e confirmação da senha diferentes. O sistema exibe a mensagem “As senhas não coincidem”.	
Linha 6: Ator informa a nova senha fora do padrão do sistema. O sistema exibe a mensagem “As senhas devem conter 8 ou mais caracteres, sendo eles, letras minúsculas, maiúsculas, números e símbolos”.	

3.1.2.2.6 Caso de Uso Alterar Endereço de entrega

Quadro 12: Descrição do Caso de Uso em Nível de Sistema - Alterar Endereço de Entrega.

Objetivo	Mudar endereço cadastrado anteriormente
Ator	Cliente

Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O endereço de entrega é modificado
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no endereço na parte superior do aplicativo.	4. Sistema exibe tela com os campos para informar o novo endereço com os campos de Rua, Cep, Número, Bairro, Cidade e UF.
5. Ator preenche as informações e toca no botão Editar Endereço.	6. Sistema exibe a mensagem “Sucesso. O Endereço de entrega foi salvo com sucesso” e retorna para a tela inicial.
Fluxo Exceção	
Linha 4: Erro ao buscar dados do endereço atual. O sistema exibe a mensagem “Não foi possível buscar as informações do endereço atual.”. Retorna para o passo 2.	
Linha 6: Ator não preenche totalmente os campos obrigatórios. O sistema exibe a mensagem “Algumas informações não foram preenchidas, é necessário informar todas os campos para finalizar o cadastro”.	
Linha 6: Ocorre um erro ao atualizar o endereço. O sistema exibe a mensagem “Ocorreu um erro ao editar o endereço, por favor tente novamente”.	

3.2 Buscou Barato Módulo Entregador

O módulo do entregador é utilizado para coleta e entrega das compras vinculadas ao usuário, na qual o mesmo, pode verificar compras destinadas a ele, as lojas em que precisa coletar os produtos das compras, assim como verificar as compras já coletadas que podem ser entregues para os clientes.

3.2.1 Requisitos funcionais e não funcionais

Os requisitos funcionais, estão resumidos no Quadro 13, são eles: 1) Cadastrar Usuário. Para ter acesso as funcionalidades do aplicativo; 2) Autenticar Usuário. É necessária para utilizar qualquer funcionalidade da aplicação. 3) Gerenciar entregas. Esse requisito permite listar compras por loja, verificar endereço de loja no mapa, concluir recebimento da compra, verificar detalhes, ver endereço de entrega no mapa e finalizar entrega vinculada ao usuário logado. 4) Alterar Senha. Esse requisito permite alterar a senha do usuário.

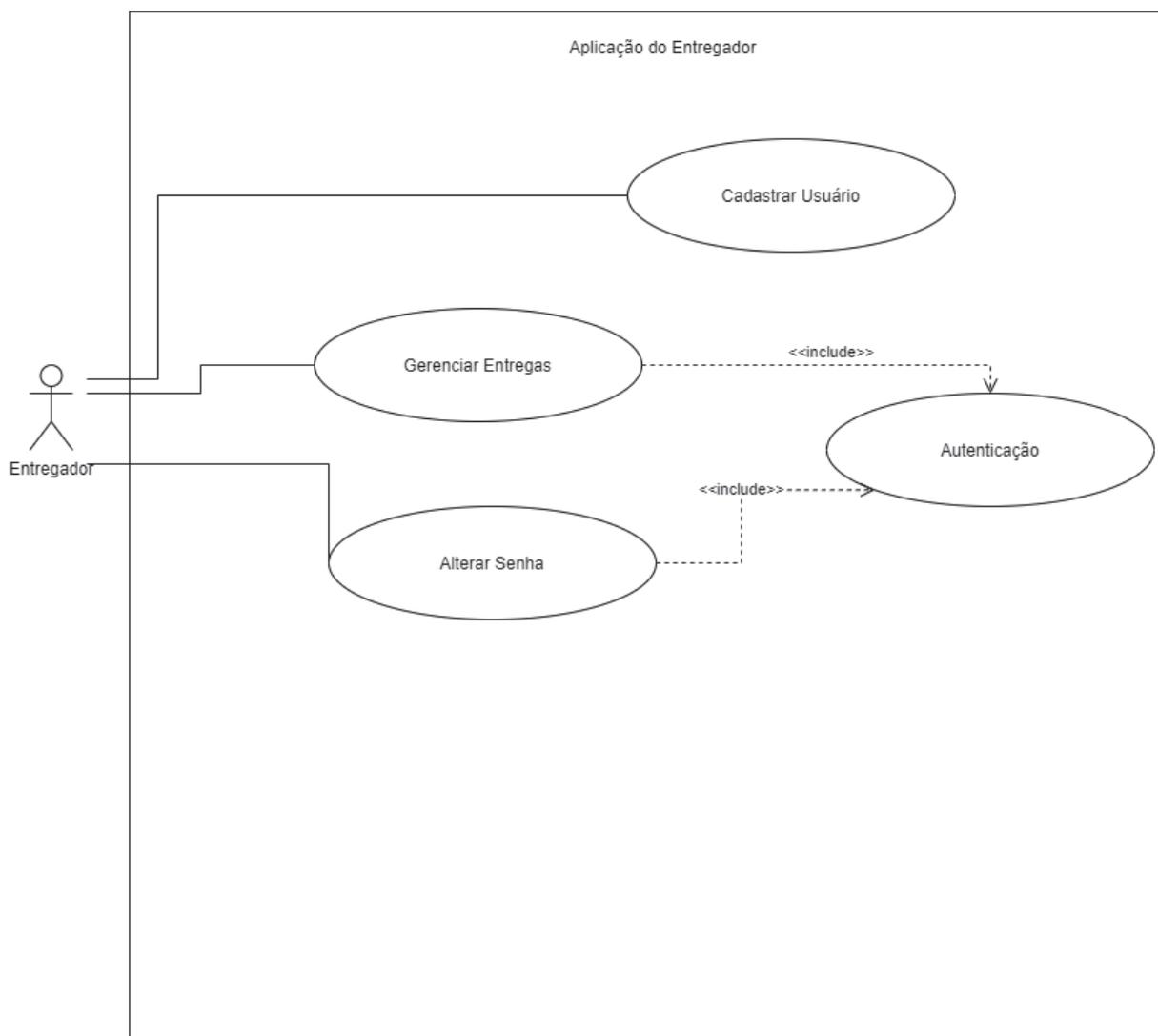
Quadro 13 - Requisitos funcionais e não funcionais da aplicação.

Requisitos Funcionais
R01: Cadastrar Usuário Entregador
R02: Autenticar Usuário
R03: Gerenciar entregas (Listar, ver endereço, concluir recebimento, finalizar entrega)
R04: Alterar Senha
Requisitos Não-Funcionais
R05: O entregador só poderá acessar as funcionalidades do aplicativo se autenticado
R06: O sistema estará disponível para as plataformas, Android, iOS e UWP
R07: O sistema deverá ser desenvolvido com a linguagem C# utilizando o Xamarin.Forms
R08: A API deverá ser desenvolvida com a linguagem Java utilizando o SpringBoot
R09: A API irá utilizar o JPA Hibernate para comunicação com o banco de dados
R10: A API deverá utilizar o banco de dados Postgresql
R11: Os sistemas precisam de acesso à internet para funcionar

3.2.2 Casos de Uso

De acordo com os requisitos citados anteriormente os casos de uso foram definidos, são eles: Cadastrar usuário; Autenticação; Gerenciar entregas; Alterar senha. Abaixo é demonstrado o diagrama de casos de uso.

Figura 5 - Diagrama de casos de uso da aplicação do entregador



3.2.2.1 Casos de Uso - Nível de Usuário

Após identificação dos requisitos necessários da aplicação, foi modelado o diagrama de casos de uso, representado na figura 5. Com ele, pode-se verificar as funcionalidades da aplicação.

O caso de uso Autenticar Usuário valida os dados de usuário e senha para permitir o acesso à aplicação. Não é possível utilizar nenhuma funcionalidade do sistema sem efetuar a autenticação. Associa-se ao requisito R02.

O caso de uso Cadastrar usuário tem como objetivo receber e gravar os dados do entregador para ser utilizado futuramente na autenticação e utilizar as funcionalidades da aplicação. É associado ao requisito R01.

O caso de uso Gerenciar Entregas, é responsável por todo o controle das entregas do usuário. Após o cliente finalizar a compra é analisado quais entregadores estão com trajetos parecidos para a coleta da compra. O entregador que tiver maior quantidade das lojas com produtos comprados pelo cliente é escolhido automaticamente, pois é o que possui o trajeto mais parecido entre os outros para efetuar a coleta. Com o caso de uso gerenciar entregas é possível o usuário verificar as lojas nas quais precisa coletar os produtos das entregas, ver a localização da loja no mapa para ajudar no processo de coleta, concluir o recebimento da compra na loja após a compra ser coletada pelo lojista, verificar os detalhes da compra, visualizar endereço de entrega e finalizar a entrega após coletar os produtos da compra nas lojas. Esse caso de uso está associado ao requisito R03.

No caso de uso Alterar Senha é possível modificar a senha do usuário, informando a senha atual e a nova. A senha deve conter no mínimo 8 caracteres, pelo menos uma letra maiúscula, pelo menos uma letra minúscula, pelo menos um número e pelo menos um caractere especial. Caso não esteja no padrão citado é mostrado uma mensagem informando que a senha precisa estar no padrão do sistema. O requisito R04 está associado a esse caso de uso.

3.2.2.2 Casos de Uso - Nível de Sistema

Serão apresentados nesta seção os casos de uso a nível de sistema. Nestes serão demonstrados o objetivo de cada função, os usuários indicados, além da pré-condição e pós-condição para realização do caso de uso. Os fluxos principais, alternativos e de exceção também serão detalhados para compreensão das interações de cada funcionalidade do sistema.

3.2.2.2.1 Caso de Uso Cadastrar Usuário

Quadro 14: Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.

Objetivo	Realizar o cadastro para acessar as funcionalidades da aplicação
Ator	Entregador

Pós-Condição	Usuário poderá acessar as funcionalidades do módulo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe tela de login.
3 .Ator toca no Botão Cadastre-se Grátis.	4. Sistema exibe a tela de cadastro.
5. Ator informa os dados cadastrais e clica em Criar Conta.	6. Sistema salva os dados e redireciona para a tela inicial.
Fluxo Exceção	
Linha 6: Dados não válidos. Sistema exibe mensagem “Algumas informações não foram preenchidas, é necessário informar todas os campos para finalizar o cadastro”, retornando ao passo 5.	

3.2.2.2.2 Caso de Uso Autenticação

Quadro 15: Descrição do Caso de Uso em Nível de Sistema - Autenticação.

Objetivo	Efetuar a autenticação para acessar as funcionalidades da aplicação
Ator	Entregador
Pós-Condição	Usuário deve ser autorizado acessar as funcionalidades do módulo do entregador
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela de login.
3. Ator informa usuário e senha e clica em Entrar.	4. Sistema verifica se os dados são válidos.
	5. Sistema permite o acesso das funcionalidades da aplicação.
Fluxo Exceção	
Linha 4: usuário ou senha informados incorretamente. Sistema exibe mensagem “Ocorreu um erro ao logar, por favor tente novamente”, retornando ao passo 3.	
Linha 4: usuário não preencheu os campos de login ou senha. Sistema exibe mensagem “Para entrar é necessário informar o campo de usuário e senha!”, retornando ao passo 3.	

3.2.2.2.3 Caso de Uso Gerenciar Entregas

Seção: Listar Compras a coletar por lojas

Quadro 16: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Listar compras a coletar por loja.

Objetivo	Listar compras de determinada loja que possuem produtos comprados pelos clientes das entregas do usuário
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	São listadas as compras da loja que precisam ser coletadas
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na loja da lista que deseja visualizar as compras.	4. Sistema exibe caixa de diálogo com as opções de Visualizar Compras e Ver no Mapa.
5. Ator toca no Botão de Visualizar Compras.	6. Sistema exibe a lista de compras a serem coletadas da loja escolhida.
Fluxo Exceção	
Linha 6: Erro ao buscar lista de compras da loja. Sistema exibe mensagem “Não foi possível buscar a lista de compras da loja.”. Retorna ao passo 4.	

Seção: Ver localização da loja no mapa

Quadro 17: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Ver localização da loja no mapa.

Objetivo	Verificar loja no mapa que possui produtos das compras vinculadas ao entregador
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	Mostra no Google Maps a localização da loja
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na loja que deseja ver a localização.	4. Sistema exibe caixa de diálogo com as opções de Visualizar Compras e Ver no Mapa.

5. Ator toca no Botão Ver no Mapa.	6. Sistema abre o Google Maps e mostra a distância e rota da localização atual do entregador para a loja.
Fluxo Alternativo	
Linha 5: Ator toca no botão Fechar. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 6: Erro ao buscar informações da loja. Sistema exibe mensagem “Ocorreu um erro ao buscar informações da loja”, retornando ao passo 5.	

Seção: Concluir Recebimento

Quadro 18: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Concluir Recebimento.

Objetivo	Concluir recebimento de produtos da loja na qual o cliente comprou
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O recebimento da compra parcial é concluído
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na loja da lista que deseja visualizar as compras.	4. Sistema exibe caixa de diálogo com as opções de Visualizar Compras e Ver no Mapa.
5. Ator toca no Botão de Visualizar Compras.	6. Sistema exibe a lista de compras a serem coletadas da loja escolhida.
7. Ator toca na compra que deseja coletar.	8. As informações da coleta, como nome do cliente, data da compra, nome da loja e lista de produtos a serem coletados.
9. Ator toca no botão Concluir Recebimento.	10. Sistema exibe mensagem “Sucesso. A compra foi coletada com sucesso!” e retorna para a tela inicial.
Fluxo Exceção	
Linha 6: Erro ao buscar lista de compras da loja. Sistema exibe mensagem “Não foi possível buscar a lista de compras da loja.”. Retorna ao passo 4.	
Linha 8: Erro ao buscar dados da compra. Sistema exibe mensagem “Não foi possível buscar os dados da compra.”. Retorna ao passo 7.	
Linha 10: Erro atualizar compra. Sistema exibe mensagem “Não foi possível realizar a atualização da compra”, retornando ao passo 9.	

Seção: Verificar Detalhes

Quadro 19: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Verificar Detalhes da compra já coletada.

Objetivo	Verificar detalhes de compra já coletada em todas as lojas
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	Verificar lista de produtos e valor da total de compra já coletada
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu Compras já Coletadas.	4. Sistema exibe lista de compras já coletadas nas lojas pelo entregador.
5. Ator toca na compra que deseja visualizar os detalhes.	6. Sistema exibe caixa de diálogo com as opções de Detalhes, Ver no Mapa e Finalizar Entrega.
7. Ator toca no botão Detalhes.	8. Sistema exibe a lista de produtos, o nome do cliente e a data da compra.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 8: Erro ao buscar dados da compra. Sistema exibe mensagem “Não foi possível buscar os dados da compra.”. Retorna ao passo 7.	

Seção: Visualizar Endereço de Entrega

Quadro 20: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Visualizar Endereço de Entrega.

Objetivo	Verificar endereço de entrega da compra já coletada
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	Mostra no Google Maps a localização do endereço de entrega da compra
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.

3. Ator toca na opção do menu Compras já Coletadas.	4. Sistema exibe lista de compras já coletadas nas lojas pelo entregador.
5. Ator toca na compra que deseja visualizar a localização.	6. Sistema exibe caixa de diálogo com as opções de Detalhes, Ver no Mapa e Finalizar Entrega.
7. Ator toca no botão Ver no Mapa.	6. Sistema abre o Google Maps e mostra a distância e rota da localização atual do entregador para o endereço de entrega.
Fluxo Alternativo	
Linha 7: Ator toca no botão Fechar. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Erro ao buscar lista de compras. Sistema exibe mensagem “Não foi possível buscar lista de compras já coletadas.”. Retorna ao passo 2.	
Linha 8: Erro ao buscar dados da compra. Sistema exibe mensagem “Não foi possível buscar os dados da compra.”. Retorna ao passo 7.	

Seção: Finalizar Entrega

Quadro 21: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Entregas - Finalizar entrega.

Objetivo	Finalizar entrega da compra do cliente
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A entrega da compra é finalizada e o status da compra atualizado para Entregue
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu Compras já Coletadas.	4. Sistema exibe lista de compras já coletadas nas lojas pelo entregador.
5. Ator toca na compra que deseja visualizar os detalhes.	6. Sistema exibe caixa de diálogo com as opções de Detalhes, Ver no Mapa e Finalizar Entrega.
7. Ator toca no botão Detalhes.	8. Sistema exibe a lista de produtos, o nome do cliente e a data da compra.
9. Ator toca no botão Finalizar Entrega.	10. Sistema finaliza a entrega e exibe a mensagem “Sucesso. A entrega foi finalizada com sucesso.”

Fluxo Alternativo
Linha 7. Ator toca no botão Fechar. Retorna para a tela anterior.
Fluxo Exceção
Linha 4: Erro ao buscar lista de compras. Sistema exibe mensagem “Não foi possível buscar lista de compras já coletadas.”. Retorna ao passo 2.
Linha 8: Erro ao buscar dados da compra. Sistema exibe mensagem “Não foi possível buscar os dados da compra.”. Retorna ao passo 7.
Linha 10. Erro atualizar compra. Sistema exibe mensagem “Não foi possível realizar a atualização da compra”, retornando ao passo 9.

3.2.2.2.4 Caso de Uso Alterar Senha

Quadro 22: Descrição do Caso de Uso em Nível de Sistema - Alterar Senha.

Objetivo	Mudar senha cadastrada anteriormente
Ator	Entregador
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A senha é modificada
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Configurações.	4. Sistema exibe tela com os campos para informar a senha atual, a nova senha e a confirmação da nova senha.
5. Ator informa a senha atual, a nova senha e a confirmação da senha e toca no botão Trocar Senha.	6. Sistema exibe a mensagem “Sucesso. A senha atual foi atualizada com sucesso” e retorna para a tela inicial.
Fluxo Exceção	
Linha 6: Ator informa a senha atual incorretamente. O sistema exibe a mensagem “A senha atual está incorreta”.	
Linha 6: Ator informa a nova senha e confirmação da senha diferentes. O sistema exibe a mensagem “As senhas não coincidem”.	
Linha 6: Ator informa a nova senha fora do padrão do sistema. O sistema exibe a mensagem “As senhas devem conter 8 ou mais caracteres, sendo eles, letras minúsculas, maiúsculas, números e símbolos”.	

3.3 Buscou Barato Módulo Lojista

A aplicação do lojista é utilizada para coleta das compras efetuadas pelos clientes dos produtos da loja, além disso, é possível gerenciar os produtos para ficarem disponíveis no

aplicativo do cliente, também existe a funcionalidade de verificar as compras que já foram coletadas e ainda não foram recolhidas pelo entregador.

3.3.1 Requisitos funcionais e não funcionais

Os requisitos funcionais, estão resumidos no Quadro 23, são eles: 1) Cadastrar Usuário Lojista. Para ter acesso as funcionalidades do aplicativo; 2) Autenticar Usuário. É necessária para utilizar qualquer funcionalidade da aplicação. 3) Gerenciar coletas. Esse requisito permite listar compras a coletar dos clientes, concluir coleta, verificar detalhes e verificar compras já coletadas. 4) Gerenciar produtos da loja. Com esse requisito é possível cadastrar, editar, alterar estoque e remover produtos da loja. 5) Alterar Senha. Esse requisito permite alterar a senha do usuário.

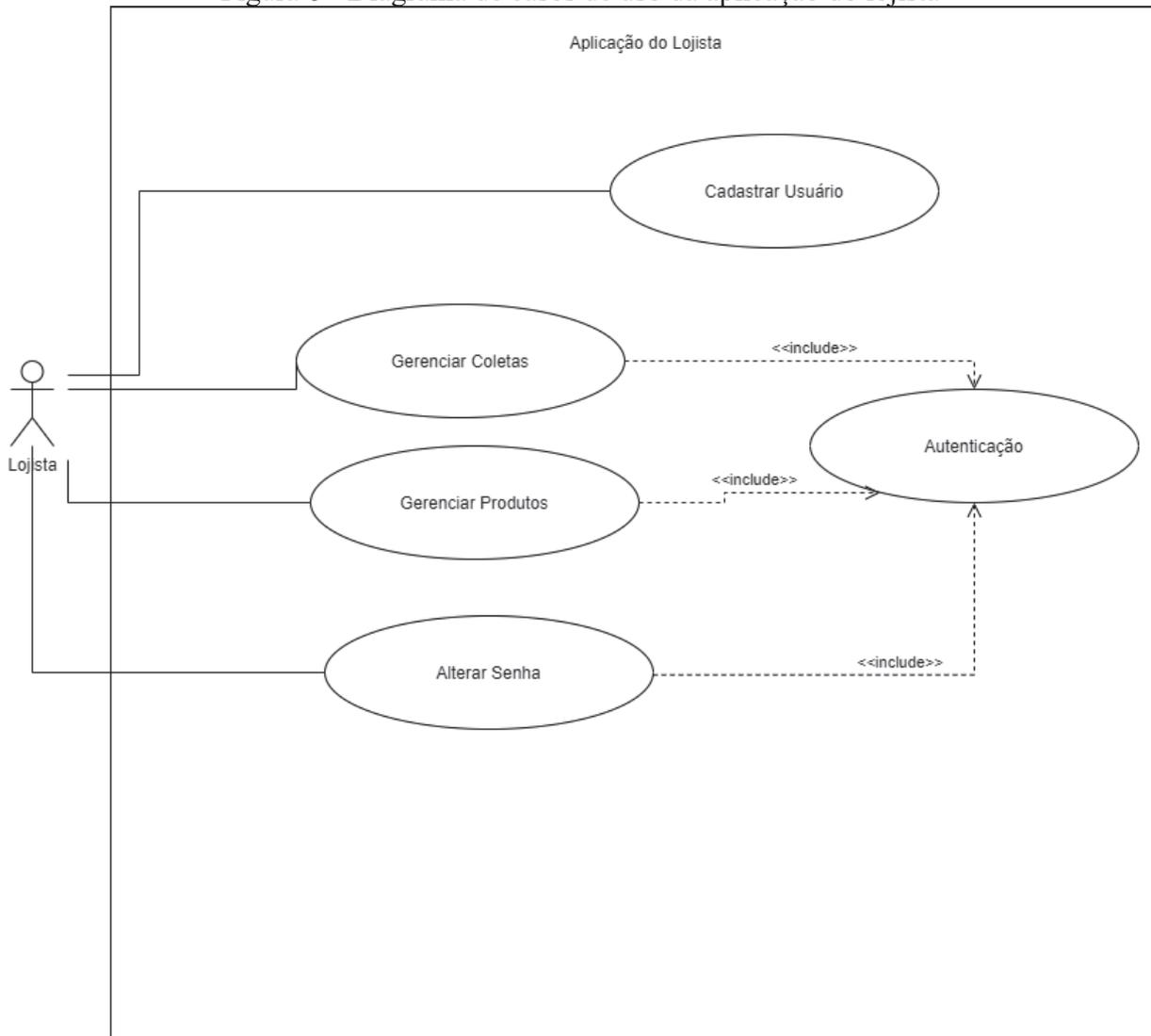
Quadro 23 - Requisitos funcionais e não funcionais da aplicação.

Requisitos Funcionais
R01: Cadastrar Usuário Lojista
R02: Autenticar Usuário
R03: Gerenciar coletas (Listar, Concluir Coleta, Detalhar e Verificar compras já coletadas)
R04: Gerenciar produtos (Cadastrar, Editar, Listar e Remover)
R05: Alterar Senha
Requisitos Não-Funcionais
R06: O lojista só poderá acessar as funcionalidades do aplicativo se autenticado
R07: O sistema estará disponível para as plataformas, Android, iOS e UWP
R08: O sistema deverá ser desenvolvido com a linguagem C# utilizando o Xamarin.Forms
R09: A API deverá ser desenvolvida com a linguagem Java utilizando o SpringBoot
R10: A API irá utilizar o JPA Hibernate para comunicação com o banco de dados
R11: A API deverá utilizar o banco de dados Postgresql
R12: Os sistemas precisam de acesso à internet para funcionar

3.3.2 Casos de Uso

De acordo com os requisitos descritos anteriormente foram estabelecidos os seguintes casos de uso: Cadastrar usuário; Autenticação; Gerenciar coletas; Gerenciar Produtos; Alterar senha. Abaixo é demonstrado o diagrama de casos de uso.

Figura 6 - Diagrama de casos de uso da aplicação do lojista



3.3.2.1 Casos de Uso - Nível de Usuário

Após análise inicial e identificação dos requisitos do aplicativo, foi modelado o diagrama de casos de uso, demonstrado na figura 6. Com ele pode-se verificar as funcionalidades da aplicação.

O caso de uso Autenticação valida os dados de usuário e senha para permitir acesso as funcionalidades do aplicativo. Não é possível utilizar nenhuma funcionalidade do sistema sem efetuar a autenticação. É associado ao requisito R02.

O caso de uso Cadastrar usuário permite receber e gravar os dados do usuário do lojista para que ele possa autenticar-se no sistema e fazer uso do mesmo. Possui o requisito R01 associado.

No caso de uso Gerenciar coletas é possível listar as compras efetuadas pelos clientes na loja do usuário, concluir a coleta depois de recolher todos os produtos da compra, verificar os detalhes de cada compra, além de verificar as compras já coletadas na loja ainda não recolhidas pelo entregador. Esse caso de uso está associado ao requisito R03.

O caso de uso Gerenciar produtos tem como objetivo criar, alterar, excluir e listar os produtos da loja que podem ser comprados pelos clientes. Na criação dos produtos, os lojistas utilizam os produtos já cadastrados na base de dados e adicionam o estoque disponível para vender no aplicativo para não haver redundâncias de produtos no sistema. O estoque dos produtos no aplicativo deve ser separado do estoque para vendas presenciais pois não há sincronização entre o sistema de automação comercial utilizado pelo lojista e o aplicativo buscou barato do lojista. Associa-se ao requisito R04.

No caso de uso Alterar Senha é possível modificar a senha do usuário, informando a senha utilizada atualmente e a nova. A senha precisa estar no padrão do aplicativo e deve conter no mínimo 8 caracteres, pelo menos uma letra maiúscula, pelo menos uma letra minúscula, pelo menos um número e pelo menos um caractere especial. Caso não esteja no padrão é mostrado uma mensagem informando que a senha precisa estar no padrão do sistema. O requisito R05 está associado a esse caso de uso.

3.3.2.2 Casos de Uso - Nível de Sistema

Serão apresentados nesta seção os casos de uso a nível de sistema. Nestes serão demonstrados o objetivo de cada função, os usuários indicados, além da pré-condição e pós-condição para realização do caso de uso. Os fluxos principais, alternativos e de exceção também serão detalhados para compreensão das interações de cada funcionalidade do sistema.

3.3.2.2.1 Caso de Uso Cadastrar Usuário

Quadro 24: Descrição do Caso de Uso em Nível de Sistema - Cadastrar Usuário.

Objetivo	Realizar o cadastro para acessar as funcionalidades da aplicação
Ator	Lojista
Pós-Condição	Usuário poderá acessar as funcionalidades do módulo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe tela de login.
3 .Ator toca no Botão Cadastre-se Grátis.	4. Sistema exibe a tela de cadastro.
5. Ator informa os dados cadastrais e clica em Criar Conta.	6. Sistema salva os dados e redireciona para a tela inicial.
Fluxo Exceção	
Linha 6: Dados não válidos. Sistema exibe mensagem “Algumas informações não foram preenchidas, é necessário informar todas os campos para finalizar o cadastro”, retornando ao passo 5.	

3.3.2.2.2 Caso de Uso Autenticação

Quadro 25: Descrição do Caso de Uso em Nível de Sistema – Autenticação.

Objetivo	Efetuar a autenticação para acessar as funcionalidades da aplicação
Ator	Lojista
Pós-Condição	Usuário deve ser autorizado acessar as funcionalidades do módulo do entregador
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela de login.
3. Ator informa usuário e senha e clica em Entrar.	4. Sistema verifica se os dados são válidos.
	5. Sistema permite o acesso das funcionalidades da aplicação.
Fluxo Exceção	
Linha 4: usuário ou senha informados incorretamente. Sistema exibe mensagem “Ocorreu um erro ao logar, por favor tente novamente”, retornando ao passo 3.	
Linha 4: usuário não preencheu os campos de login ou senha. Sistema exibe mensagem “Para entrar é necessário informar o campo de usuário e senha!”, retornando ao passo 3.	

3.3.2.2.3 Caso de Uso Gerenciar Coletas

Seção: Listar Compras efetuadas pelos clientes na loja

Quadro 26: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Listar Compras efetuadas pelos clientes na loja.

Objetivo	Listar compras efetuadas pelos clientes na loja
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	São listadas as compras da loja que precisam ser coletadas
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Início.	4. Sistema a lista de compras que precisam ser coletadas.

Seção: Verificar detalhes da compra

Quadro 27: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Verificar detalhes da compra.

Objetivo	Verificar detalhes e produtos da compra
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	É possível verificar o detalhamento da compra
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na compra que deseja finalizar a coleta.	4. Sistema os detalhes da compra e a lista de produtos.
Fluxo Exceção	

Linha 4: Ocorre um erro ao buscar as informações da compra. Sistema exibe a mensagem “Não foi possível buscar os detalhes da compra”. Retorna ao passo 3.

Seção: Concluir coleta

Quadro 28: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Concluir Coleta.

Objetivo	Concluir coleta de compra efetuada na loja
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A compra fica disponível para o entregador recolher
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na compra que deseja finalizar a coleta.	4. Sistema os detalhes da compra e a lista de produtos.
5. Ator toca no botão Concluir coleta.	6. Sistema exibe a mensagem “Sucesso. A compra foi coletada com sucesso.”
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar as informações da compra. Sistema exibe a mensagem “Não foi possível buscar os detalhes da compra”. Retorna ao passo 3.	
Linha 6: Ocorre um erro na atualização da compra. Sistema exibe a mensagem “Não foi possível realizar a atualização da compra!”.	

Seção: Listar Compras já coletadas

Quadro 29: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Coletas - Listar Compras efetuadas pelos clientes na loja já coletadas.

Objetivo	Listar compras efetuadas pelos clientes na loja já recolhidas pelo lojista
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	São listadas as compras da loja que já foram coletadas
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.

3. Ator toca no item do menu Compras Coletadas.	4. Sistema exibe a lista de compras que precisam já coletadas.
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar as compra coletadas. Sistema exibe a mensagem “Não foi possível buscar a lista de compras coletadas.”. Retorna ao passo 2.	

3.3.2.2.4 Caso de Uso Gerenciar Produtos

Seção: Listar Produtos cadastrados da loja

Quadro 30: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Listar produtos cadastrados da loja.

Objetivo	Listar produtos da loja cadastrados no sistema
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	São listados os produtos da loja cadastrados no sistema
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Produtos da loja.	4. Sistema exibe a lista de produtos cadastrados da loja.
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar a lista de produtos cadastrados. Sistema exibe a mensagem “Não foi possível buscar a lista de produtos cadastrados.”. Retorna ao passo 2.	

Seção: Cadastrar produtos

Quadro 31: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Cadastrar Produtos.

Objetivo	Efetuar cadastro de produto da loja para ficar disponível no módulo do cliente
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema.
Pós-Condição	O produto é cadastrado e possível de ser comprado no aplicativo do cliente

Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Produtos da loja.	4. Sistema exibe a lista de produtos cadastrados da loja.
5. Ator toca no botão flutuante +.	6. Sistema exibe tela de cadastro com os dados que precisam ser preenchidos.
7. Ator preenche as informações e toca no botão Adicionar Produto na Loja.	8. Sistema exibe mensagem “Sucesso. O produto foi cadastrado com sucesso” e retorna para a lista de produtos.
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar a lista de produtos cadastrados. Sistema exibe a mensagem “Não foi possível buscar a lista de produtos cadastrados.”. Retorna ao passo 2.	
Linha 8: Ator não preenche todos os dados solicitados. Sistema exibe a mensagem “Algumas informações não foram preenchidas, é necessário informar todos os campos para finalizar o cadastro”.	
Linha 8: Ocorre um erro ao cadastrar produto na loja. Sistema exibe a mensagem “Ocorreu um erro ao tentar cadastrar o produto na loja”.	

Seção: Editar produtos

Quadro 32: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Editar Produtos.

Objetivo	Efetuar edição de produto da loja para ficar disponível no módulo do cliente
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O produto é alterado e possível de ser comprado no aplicativo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Produtos da loja.	4. Sistema exibe a lista de produtos cadastrados da loja.

5. Ator toca no produto da lista.	6. Sistema exibe caixa de diálogo com as opções de Editar Produto, Alterar Estoque e Remover Produto.
7. Ator toca no botão de Editar Produto.	8. Sistema exibe tela de edição com os dados que precisam ser preenchidos.
9. Ator preenche os dados e toca no botão Editar produto da loja.	10. Sistema exibe mensagem “Sucesso. O produto foi alterado com sucesso” e retorna para a lista de produtos.
Fluxo Alternativo	
Linha 7: Ator toca no botão para fechar a caixa de diálogo. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar a lista de produtos cadastrados. Sistema exibe a mensagem “Não foi possível buscar a lista de produtos cadastrados.”. Retorna ao passo 2.	
Linha 10: Ator não preenche todos os dados solicitados. Sistema exibe a mensagem “Algumas informações não foram preenchidas, é necessário informar todos os campos para finalizar a edição”.	
Linha 10: Ocorre um erro ao editar produto na loja. Sistema exibe a mensagem “Ocorreu um erro ao tentar editar o produto na loja”.	

Seção: Remover produtos

Quadro 33: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Remover produtos.

Objetivo	Remover produto da loja
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O produto é removido da loja e fica indisponível no aplicativo do cliente
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Produtos da loja.	4. Sistema exibe a lista de produtos cadastrados da loja.
5. Ator toca no produto da lista.	6. Sistema exibe caixa de diálogo com as opções de Editar Produto, Alterar Estoque e Remover Produto.

7. Ator toca no botão de Remover Produto.	8. Sistema exibe caixa de diálogo para confirmar se deseja remover o produto.
9. Ator toca no botão Sim.	10. Sistema exibe mensagem “Sucesso. O produto foi removido com sucesso” e retorna para a lista de produtos.
Fluxo Alternativo	
Linha 7: Ator toca no botão para fechar a caixa de diálogo. Retorna para a tela anterior.	
Linha 9: Ator toca no botão Não. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar a lista de produtos cadastrados. Sistema exibe a mensagem “Não foi possível buscar a lista de produtos cadastrados.”. Retorna ao passo 2.	
Linha 10: Ocorre um erro ao remover produto da loja. Sistema exibe a mensagem “Não foi possível remover o produto da loja”.	

Seção: Editar estoque de produtos

Quadro 34: Descrição do Caso de Uso em Nível de Sistema - Gerenciar Produtos - Editar Estoque dos produtos.

Objetivo	Editar estoque de produtos
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	O estoque do produto é alterado
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca no item do menu Produtos da loja.	4. Sistema exibe a lista de produtos cadastrados da loja.
5. Ator toca no produto da lista.	6. Sistema exibe caixa de diálogo com as opções de Editar Produto, Alterar Estoque e Remover Produto.
7. Ator toca no botão Alterar Estoque.	8. Sistema exibe caixa de diálogo para informar o novo estoque do produto.

9. Ator informa a nova quantidade e toca no botão Editar Quantidade.	10. Sistema exibe mensagem “Sucesso. O estoque foi atualizado com sucesso” e retorna para a lista de produtos.
Fluxo Alternativo	
Linha 7: Ator toca no botão para fechar a caixa de diálogo. Retorna para a tela anterior.	
Linha 9: Ator toca no botão para fechar a caixa de diálogo. Retorna para a tela anterior.	
Fluxo Exceção	
Linha 4: Ocorre um erro ao buscar a lista de produtos cadastrados. Sistema exibe a mensagem “Não foi possível buscar a lista de produtos cadastrados.”. Retorna ao passo 2.	
Linha 10: Ocorre um erro ao editar estoque do produto. Sistema exibe a mensagem “Ocorreu um erro ao tentar atualizar o estoque do produto”.	

3.3.2.2.5 Caso de Uso Alterar Senha

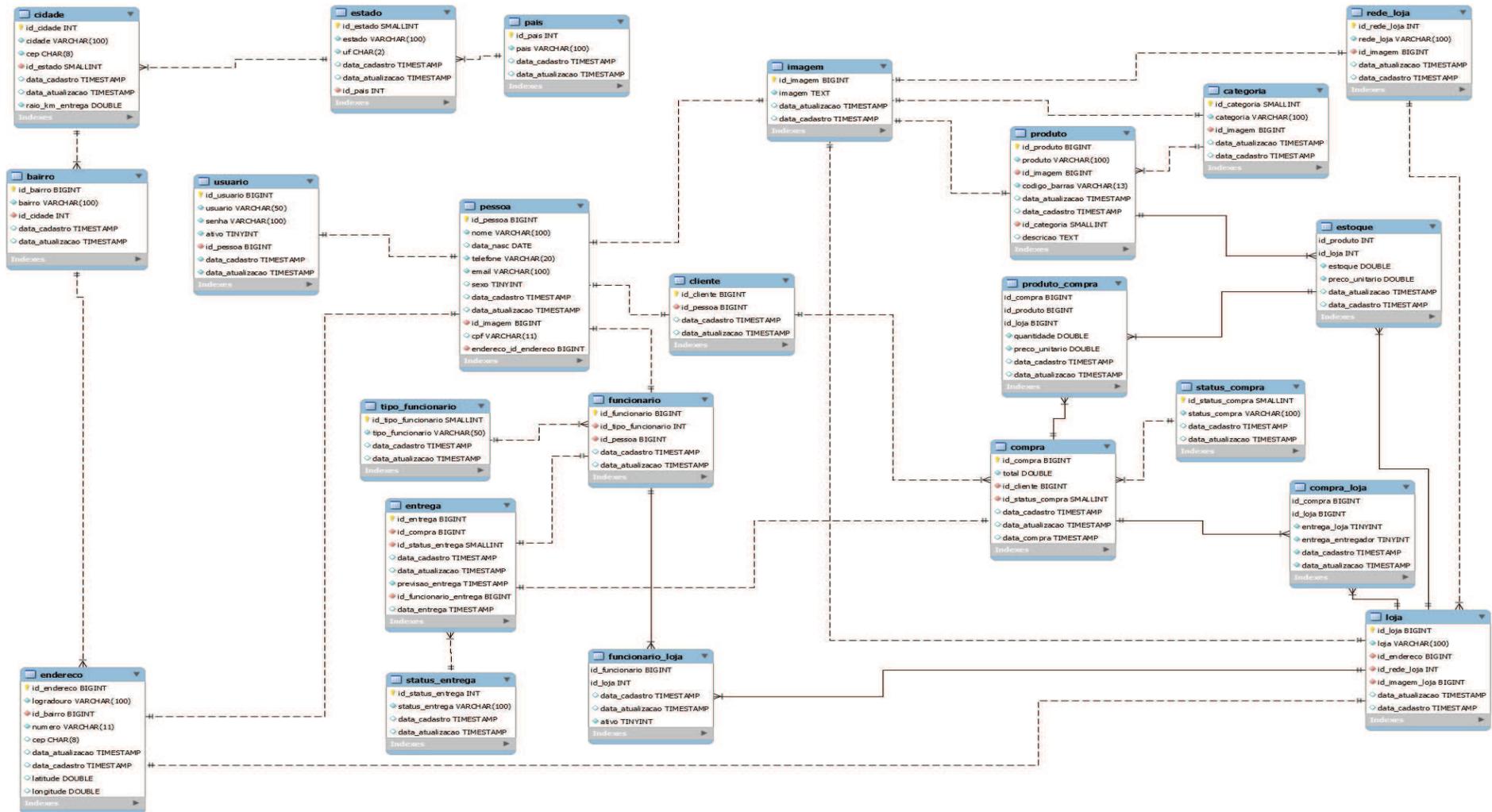
Quadro 35: Descrição do Caso de Uso em Nível de Sistema - Alterar Senha.

Objetivo	Mudar senha cadastrada anteriormente.
Ator	Lojista
Pré-Condição	O Ator precisa estar autenticado no sistema
Pós-Condição	A senha do usuário é modificada
Fluxo Principal	
Ação do Ator	Resposta do Sistema
1. Ator abre a aplicação.	2. Sistema exibe a tela inicial.
3. Ator toca na opção do menu de Configurações.	4. Sistema exibe tela com os campos para informar a senha atual, a nova senha e a confirmação da nova senha.
5. Ator informa a senha atual, a nova senha e a confirmação da senha e toca no botão Trocar Senha.	6. Sistema exibe a mensagem “Sucesso. A senha atual foi atualizada com sucesso” e retorna para a tela inicial.
Fluxo Exceção	
Linha 7: Ator informa a senha atual incorretamente. O sistema exibe a mensagem “A senha atual está incorreta”.	
Linha 7: Ator informa a nova senha e confirmação da senha diferentes. O sistema exibe a mensagem “As senhas não coincidem”.	
Linha 7: Ator informa a nova senha fora do padrão do sistema. O sistema exibe a mensagem “As senhas devem conter 8 ou mais caracteres, sendo eles, letras minúsculas, maiúsculas, números e símbolos”.	

3.4 Modelagem de Dados

Após a definição dos requisitos e casos de uso, foi criado o projeto do esquema lógico de dados, em seguida foram detalhadas as entidades e a definição dos seus atributos. A figura 7 mostra o esquema lógico da aplicação.

Figura 7 Esquema de dados do sistema.



3.4.1 Componentes do modelo de dados

Nesta seção serão apresentadas as entidades do sistema e o que representam, assim como os seus atributos. As entidades identificadas para elaboração da base de dados estão descritas abaixo no quadro 36.

Quadro 36: Definição das tabelas.

Tabela	Descrição
Bairro	A entidade Bairro representa o bairro do Endereço. Um Bairro pertence a uma Cidade, e um Bairro pode ter vários endereços vinculados a ele.
Categoria	Essa entidade representa a Categoria dos Produtos. Uma categoria pode ter vários produtos.
Cidade	Representa a Cidade do Endereço. Uma ou mais cidades podem pertencer a um Estado, e uma Cidade pode ter vários Bairros.
Cliente	Essa entidade representa o Cliente apto a fazer compras no aplicativo do Cliente. Quando é criado um novo Usuário no aplicativo do Cliente também é criado um Cliente.
Compra	A entidade Compra representa uma Compra efetuada por um Cliente em uma ou mais lojas com uma Entrega vinculada. Uma compra possui uma entrega vinculada, e possui também um status utilizado para ser tratado de forma correta por Entregadores, Lojistas e Clientes..
Compra_Loja	Representa a relação de Compra e Loja, possui atributos de controle para Coleta e Recebimento de Lojistas e Entregadores.
Endereco	As entidades, Loja e Pessoa possuem um endereço vinculado a elas. Possui as informações de latitude e longitude inseridas no momento do cadastro do usuário que são utilizadas no sistema para geolocalização. Um endereço pertence a um Bairro..
Entrega	Essa entidade representa a Entrega de uma Compra. A Entrega também contém um status, utilizado para tratar a entrega corretamente pelo Entregador e Cliente..
Estado	A entidade Estado representa o Estado da entidade Endereço. Um estado pertence a um

	Pais e um Estado pode conter várias cidades.
Estoque	Representa a relação da tabela Produto com a tabela Loja, nela contém informações como o estoque desse produto e o preço unitário dele na Loja.
Funcionario	Representa um Funcionário, que pode ser Funcionário de loja ou Entregador. Um funcionario contém uma pessoa associada e é de um tipo específico.
Funcionario_Loja	Representa a relação das tabelas Funcionario e Loja. Um funcionário que coleta os produtos da loja é representado por essa tabela. Possui o atributo ativo para representar se o vínculo desse Funcionário com determinada Loja está ativado e manter um histórico.
Imagem	Armazena as imagens do sistema. As imagens são convertidas em Base64 e armazenadas nessa tabela. Produtos, Lojas, Pessoas, Categorias e Redes de Lojas contém imagens vinculadas a elas.
Loja	A entidade Loja, representa a loja do sistema, ela pode conter vários produtos. Possui um Endereço, uma imagem e pertence a uma Rede de Lojas.
Pais	Representa o País no banco do sistema, ele pode ter vários Estados.
Pessoa	Representa a Pessoa e possui os atributos dos dados pessoais dos clientes e funcionários. Uma pessoa tem uma imagem, um usuário, um endereço e pode ter um cliente e um funcionário vinculado a ela.
Produto	Essa entidade representa o Produto do sistema e pertence a uma categoria e pode estar em uma ou mais Lojas. Além disso, possui uma imagem..
Produto_Compra	Representa um produto da compra de um Cliente. Pertence a uma Loja através da tabela Estoque. Possui os atributos de quantidade de produtos comprados e o preço da unidade no momento da compra.
Rede_Loja	Representa as redes de lojas, uma rede de lojas pode ter várias lojas e uma loja pertence a uma rede lojas. Também possui uma imagem vinculada.
Status_Compra	Representa o status que uma compra pode ter. São eles: Em Aberto, Aguardando Pagamento, Entregue, Aguardando Início

	Coleta e Em Coleta.
Status_Entrega	Representa o status que uma entrega pode ter. São eles: Em Aberto, Em Coleta e Entregue.
Tipo_Funcionario	Representa os tipos de funcionários. São eles: Entregador e Coletor.
Usuario	A entidade usuário representa aquele que terá acesso às funções das aplicações. Pode ser um entregador, lojista ou cliente. Um usuário possui uma pessoa vinculado a ele..

Após a definição das tabelas foram definidos os atributos e a representação de cada um. O quadro 37 descreve os atributos das tabelas do sistema.

Quadro 37: Definição dos atributos.

Tabela	Atributo	Descrição dos atributos
Bairro	Id_Bairro Bairro Id_cidade Data_Cadastro Data_Atualizacao	Código identificador do bairro Nome do bairro Identificação da cidade relacionada Data de cadastro do bairro Data de atualização do bairro
Categoria	Id_Categoria Categoria Id_Imagem Data_Cadastro Data_Atualizacao	Código identificador da categoria Nome da categoria Identificação imagem relacionada Data de cadastro da categoria Data de atualização da categoria
Cidade	Id_Cidade Cidade Cep Id_Estado Data_Cadastro Data_Atualizacao Raio_KM	Código identificador da cidade Nome da Cidade Cep da cidade Identificador de estado relacionado Data de cadastro da cidade Data de atualização da cidade Raio de busca de produtos na cidade
Cliente	Id_Cliente Id_Pessoa Data_Cadastro Data_Atualizacao	Código identificador do cliente Identificador da pessoa relacionada Data de cadastro do cliente Data de atualização do cliente
	Id_Compra Total	Código identificador da compra

Compra	Id_Cliente Id_Status_Compra Data_Compra Data_Cadastro Data_Atualizacao	Valor total da compra Identificador do cliente relacionado Status da compra relacionado Data de finalização da compra Data de cadastro da compra Data de atualização da compra
Compra_Loja	Id_Loja Id_Compra Entrega_Loja Entrega_Entregador Data_Cadastro Data_Atualizacao	Identificador da loja relacionada Identificador da compra relacionada Flag de entrega para entregador Flag de entrega para cliente Data de cadastro Data de atualização
Endereco	Id_Endereco Logradouro Id_Bairro Numero Cep Data_Cadastro Data_Atualizacao Latitude Longitude	Código identificador de endereço Nome logradouro Identificador de bairro relacionado Número do logradouro Cep do endereço Data de cadastro do endereço Data de atualização do endereço Latitude do endereço Longitude do endereço
Entrega	Id_Entrega Id_Compra Id_Status_Entrega Data_Cadastro Data_Atualizacao Previsao_Entrega Data_Entrega Id_Funcionario_Entrega	Código identificador da entrega Identificador da compra relacionada Identificador de status de entrega Data de cadastro da entrega Data de atualização da entrega Previsão da entrega Data da entrega Identificador funcionário entregador
Estado	Id_Estado Estado UF Data_Cadastro Data_Atualizacao Id_Pais	Código identificador do estado Nome do estado Sigla UF Data de cadastro do Estado Data de atualização do Estado Identificador de país relacionado
	Id_Produto Id_Loja Estoque	Identificador de produto relacionado Identificador de loja

Estoque	Preco_Unitario Data_Cadastro Data_Atualizacao	relacionada Quantidade de produto em estoque Preço da unidade do produto na loja Data de cadastro Data de atualização
Funcionario	Id_Funcionario Id_Pessoa Id_Tipo_Funcionario Data_Cadastro Data_Atualizacao	Código identificador do funcionário Identificador da pessoa relacionada Identificador tipo relacionado Data de cadastro do funcionário Data de atualização do funcionário
Funcionario_Loja	Id_Loja Id_Funcionario Data_Cadastro Data_Atualizacao Ativo	Identificador da loja relacionada Identificador funcionário relacionado Data de cadastro Data de atualização Flag para identificar se está ativo
Imagem	Id_Imagem Descricao Imagem Data_Cadastro Data_Atualizacao	Código identificador da imagem Descrição da imagem Imagem no formato base64 Data de cadastro da imagem Data de atualização da imagem
Loja	Id_Loja Loja Id_Rede_Loja Id_Endereco Id_Imagem Data_Cadastro Data_Atualizacao	Código identificador da loja Nome da loja Identificador rede de lojas relacionada Identificador de endereço da loja Identificador de imagem relacionada Data de cadastro da imagem Data de atualização da imagem
Pais	Id_Pais Pais Data_Cadastro Data_Atualizacao	Código identificador do país Nome do país Data de cadastro do país Data de atualização do país
	Id_Pessoa Nome Data_Nasc Email	Código identificador da pessoa Nome da pessoa Data de nascimento da pessoa E-mail da pessoa

Pessoa	Sexo Data_Cadastro Data_Atualizacao Id_Imagem Cpf Id_Endereco Telefone	Sexo da pessoa Data de cadastro da pessoa Data de atualização da pessoa Identificador da imagem relacionada CPF da pessoa Identificador do endereço relacionado Telefone da pessoa
Produto	Id_Produto Produto Codigo_Barras Descricao Id_Imagem Data_Cadastro Data_Atualizacao Id_Categoria	Código identificador do produto Nome do produto Código de barras do produto Descrição do produto Identificador da imagem relacionada Data de cadastro do produto Data de atualização do produto Identificador da categoria relacionada
Produto_Compra	Id_Loja Id_Compra Id_Produto Data_Cadastro Data_Atualizacao Quantidade Preco_Unitario	Identificador da loja relacionada Identificador da compra relacionada Identificador do produto relacionado Data de cadastro Data de atualização Quantidade do produto na compra Preço unitário no momento da compra
Rede_Loja	Id_Rede_Loja Rede_Loja Id_Imagem Data_Cadastro Data_Atualizacao	Código identificador da rede de lojas Nome da rede de lojas Identificador da imagem relacionada Data de cadastro da rede de lojas Data de atualização da rede de lojas
Status_Compra	Id_Status_Compra Status_Compra Data_Cadastro Data_Atualizacao	Identificador do status da compra Nome do status da compra Data do cadastro Data de atualização
Status_Entrega	Id_Status_Entrega Status_Entrega	Identificador do status da entrega

	Data_Cadastro Data_Atualizacao	Nome do status da entrega Data do cadastro Data da atualização
Tipo_Funcionario	Id_Tipo_Funcionario Tipo_Funcionario Data_Cadastro Data_Atualizacao	Identificador do tipo de funcionário Nome do tipo de funcionário Data do cadastro Data da atualização
Usuario	Id_Usuario Usuario Senha Id_Pessoa Ativo Data_Cadastro Data_Atualizacao	Código identificador de usuário Usuário Senha do usuário Identificador da pessoa relacionada Flag para identificar se está ativo Data de cadastro de usuário Data de atualização do usuário

4 BUSCOU BARATO

Nesta seção será apresentada a aplicação desenvolvida. As telas serão exibidas e o fluxo de utilização. O sistema está dividido em três módulos, cliente, entregador e lojista, então, apresentaremos cada um por seção.

4.1. Buscou Barato Cliente

Tela Inicial

A tela inicial do aplicativo do cliente é apresentada na figura 8. Se o cliente não estiver logado é mostrado os produtos das lojas próximas baseado na localização atual do dispositivo. Nesse caso somente é possível visualizar os produtos além de visualizar produtos de determinada categoria tocando em alguma das opções da lista horizontal acima da lista de produtos. Caso o usuário não tenha realizado o login e tentar acessar alguma funcionalidade não permitida o aplicativo mostra um *pop-up* informando que é necessário efetuar o login na aplicação, essa mensagem é representada na figura 9.

As funcionalidades após efetuar a autenticação são as de Gerenciar os produtos do carrinho de compras, Gerenciar compras, o *pop-up* para pesquisa, que pode ser aberto ao tocar no botão da lupa no canto superior direito, outra funcionalidade é a de abrir a tela para mudança

de endereço tocando no endereço acima da lista de categorias. Tocando em algum dos produtos da lista é aberto a tela para adição do produto no carrinho de compras. Existe também o botão de carrinho de compras em cada item da lista que possui o mesmo comportamento do toque no produto da lista, apenas com a diferença que nesse caso é aberto um *pop-up* para adicionar o produto no carrinho diferente do toque no item da lista que abre uma nova tela. Por fim existe um botão flutuante na parte inferior direita que ao ser tocado leva para a tela do carrinho de compras.

Figura 8 - Tela inicial do aplicativo sem efetuar o login.



Figura 9 - Mensagem para informar ao usuário que o login precisa ser efetuado.

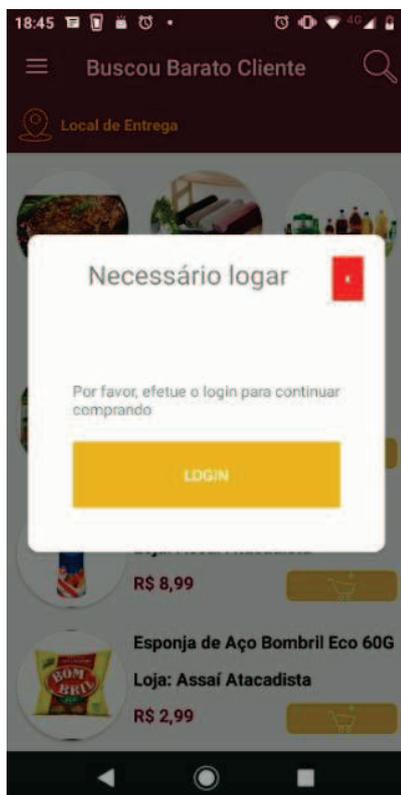


Figura 10 - Tela inicial do aplicativo com login realizado.



Menu Lateral do Aplicativo

O menu lateral do aplicativo possui dois comportamentos, dependendo se o usuário tenha efetuado a autenticação ou não. Caso a autenticação não foi realizada apenas as funcionalidades de tela inicial, login e sair do aplicativo ficam disponíveis como mostra na figura 11. Se a autenticação foi realizada as funcionalidades de tela inicial, compras, formas de Pagamento, Configurações e Sair ficam disponíveis como é demonstrado na figura 12.

Figura 11 - Menu lateral sem efetuar o login.

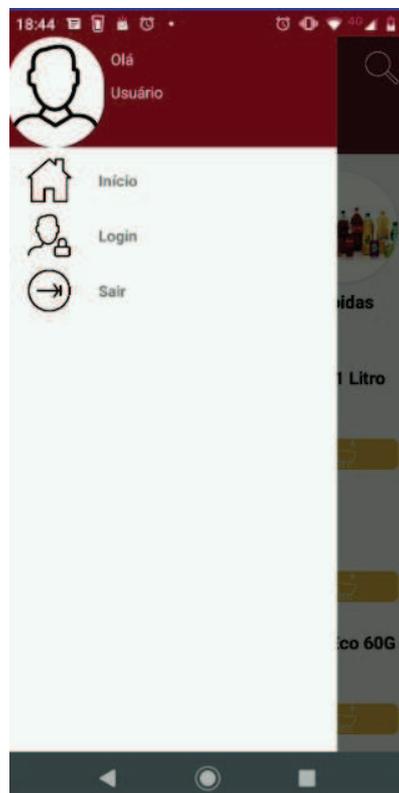
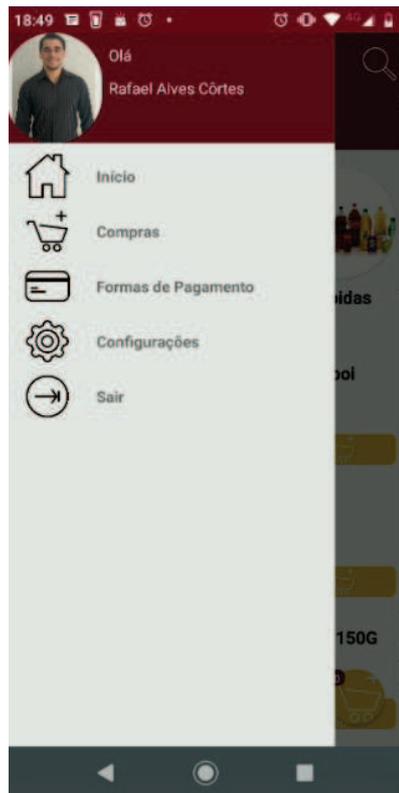


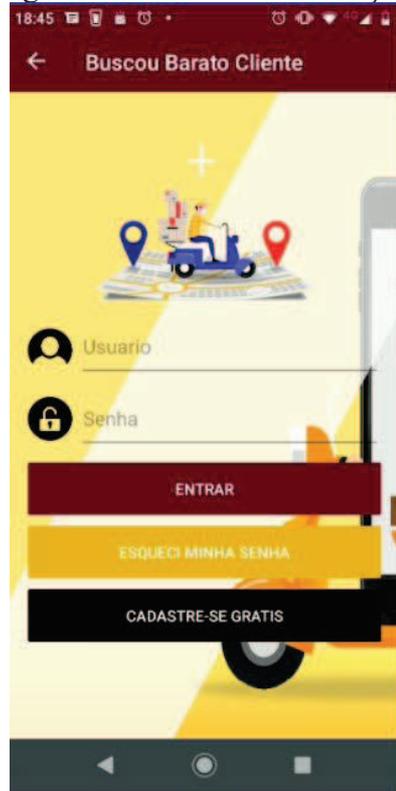
Figura 12 - Menu lateral com login realizado.



Tela de Autenticação

As principais funcionalidades do sistema somente estarão disponíveis após a autenticação ser efetuada. Assim, o cliente cadastrado anteriormente pode realizar a autenticação e utilizar todas as funcionalidades disponíveis do aplicativo. Os dados solicitados são o Usuário e Senha. Após isso é necessário tocar no botão Entrar. Outra funcionalidade disponível é a de abrir a tela para cadastrar um novo usuário tocando no botão Cadastre-se Grátis. Para acessar a tela de autenticação é necessário tocar na opção Login do menu lateral, essa opção fica disponível somente quando não há usuário logado no aplicativo. Essa tela é demonstrada na figura 13.

Figura 13 - Tela de autenticação.



Tela de Cadastro

Para utilizar todas as funcionalidades da aplicação é necessário criar uma nova conta no aplicativo cliente caso ainda não a tenha. Durante o cadastro alguns dados são solicitados, após o cadastro é possível realizar compras e alterar dados informados no momento do cadastro como mudança de senha e endereço para entrega das compras. Essa tela é representada na figura 14.

Figura 14 - Tela de cadastro.

The image displays two side-by-side screenshots of a mobile application's registration screen, titled "Buscou Barato Cliente".

The left screenshot shows the "Informações da Conta" (Account Information) section. It features a profile icon placeholder and several input fields: "Nome Completo", "Usuario", "Senha", "Confirmação de Senha", "Telefone", and "Email".

The right screenshot shows the "Endereço" (Address) section. It includes input fields for "Logradouro", "CEP", "Nº", "Bairro", "UF", and "Cidade". At the bottom of this section, there is a green button labeled "CRIAR CONTA".

Tela de pesquisa de produtos

Aqui é possível pesquisar os produtos das lojas, para chegar nessa tela basta tocar na lupa no canto superior direito da tela inicial. Para realizar a pesquisa basta informar o nome do produto. Com mais de quatro caracteres digitados o aplicativo traz sugestões de produtos das lojas próximas para pesquisa, para finalizar a busca basta tocar no botão Pesquisar. Esse *popup* está representado na figura 15.

Figura 15 - *Pop-up* de pesquisa.



Tela de mudança de endereço

Essa tela tem como objetivo a edição do endereço do cliente anteriormente cadastrado. Existe algumas validações no campo de Cep, nesse dado há uma busca após informar o Cep e os dados do endereço são automaticamente preenchidos. Se o usuário informar um Cep inexistente a aplicação irá exibir uma mensagem informando que o Cep não foi encontrado (ver figura 17). Após o preenchimento dos dados é necessário tocar no botão Editar endereço. A figura 16 representa essa tela.

Figura 16 - Tela de mudança de endereço de entrega.



Figura 17 - Mensagem de cep não encontrado.



Tela de mudança de senha

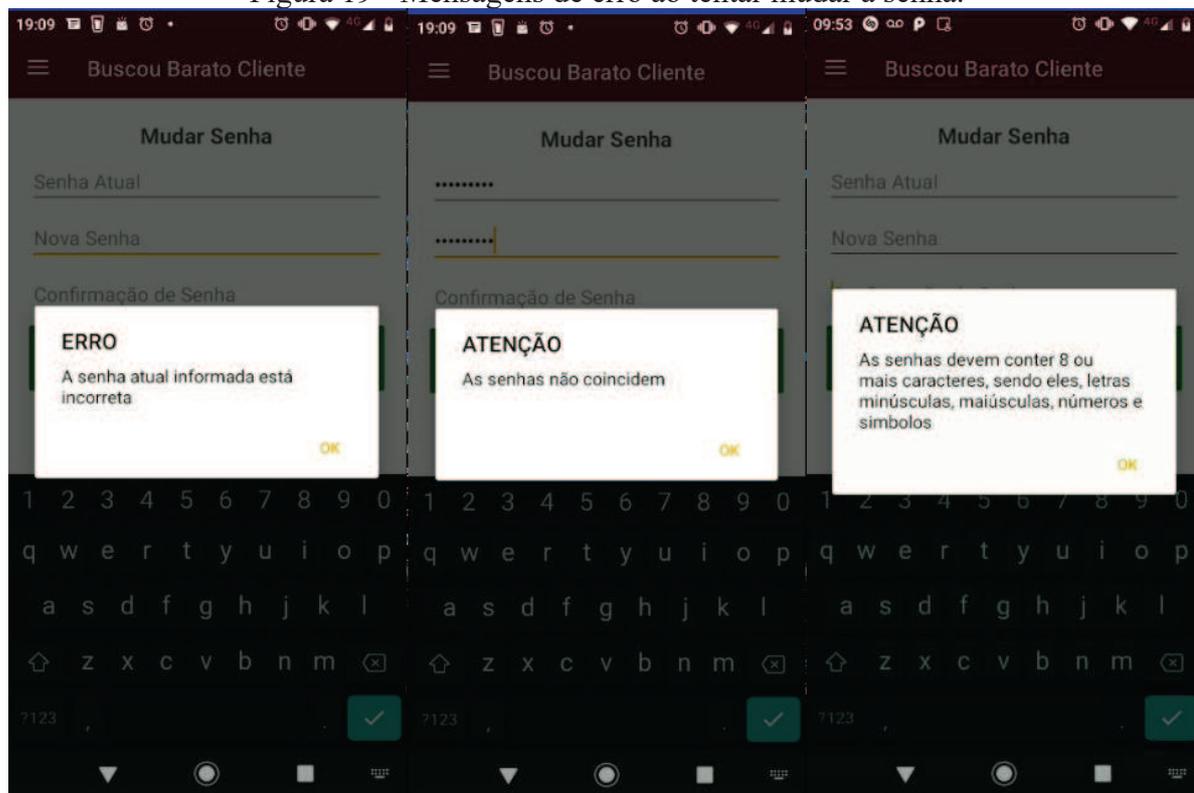
A funcionalidade de mudança de senha permite ao usuário modificar a senha atual da sua conta para uma nova. A tela de mudança de senha representada na figura 18 possui algumas validações, a senha atual deve estar correta, a nova senha deve ser igual a confirmação da senha e a nova deve estar no padrão do sistema. O padrão exige que a senha deve conter no mínimo 8 caracteres, e pelo menos, uma letra maiúscula, uma letra minúscula, um caractere especial e um número. É exibido uma mensagem para qualquer uma das validações caso os dados informados pelo usuário não estejam em conformidade com as validações exigidas, essas mensagens são representadas na figura 19.

Figura 18 - Tela de mudança de senha.



A imagem mostra a interface de usuário de um aplicativo móvel. No topo, há uma barra de status com o horário 19:08 e ícones de bateria, sinal e Wi-Fi. Abaixo, uma barra de navegação azul com o ícone de menu e o texto "Buscou Barato Cliente". O conteúdo principal da tela é branco e contém o título "Mudar Senha" em azul. Há três campos de entrada de texto: "Senha Atual", "Nova Senha" e "Confirmação de Senha". Abaixo dos campos, há um botão verde com o texto "TROCAR SENHA" em branco. Na base da tela, há uma barra de navegação padrão do Android com ícones de voltar, home e avançar.

Figura 19 - Mensagens de erro ao tentar mudar a senha.



Tela para adicionar produto no carrinho de compras

Nessa tela é possível adicionar o produto no carrinho de compras do usuário. É exibido o produto que escolhido da lista na tela inicial e a também é exibida a loja da qual ele pertence. O usuário escolhe a quantidade do produto que deseja comprar e toca no botão do carrinho para efetuar a adição do produto no carrinho de compras. A partir da tela inicial o usuário pode adicionar o produto de duas formas, tocando no produto da lista que deseja adicionar e é direcionado para a tela de cadastro, representada na figura 20, ou o usuário pode tocar no botão do carrinho do item da lista e abre um *pop-up* que possui o mesmo comportamento da tela de adição de produtos no carrinho, representado na figura 21.

Figura 20 - Tela para adicionar produto ao carrinho.



Figura 21 - *Pop-up* para adicionar produto ao carrinho.



Tela do carrinho de compras

Essa tela exibe todos os produtos adicionados no carrinho pelo usuário, a quantidade e o valor total dos produtos. Representada na figura 22, essa tela também possui as opções para editar e remover cada item do carrinho assim como finalizar a compra que dará início ao processo de coleta e entrega pelo lojista e entregador respectivamente. Para remover um item do carrinho é necessário tocar no botão remover do item que possui um ícone “-”, após o usuário tocar nesse botão o aplicativo pede uma confirmação representado na figura 23. A edição de produtos representada na figura 24 possibilita o usuário alterar a quantidade do produto comprado.

Figura 22 - Tela de carrinho de compras.



Figura 23 - Mensagem de confirmação para remover produto do carrinho.



Figura 24 - Pop-up para edição do produto no carrinho de compras.



Tela de histórico de compras

Funcionalidade permite o usuário verificar todas as compras já efetuadas, com ela é possível visualizar os detalhes da compra, rastrear a compra se estiver durante a coleta e recriar a compra caso a compra já tenha sido entregue. Essa tela é representada na figura 25, quando o usuário toca na compra é exibido o *pop-up* com as opções para realizar com a compra escolhida, esse *pop-up* é exibido na figura 26.

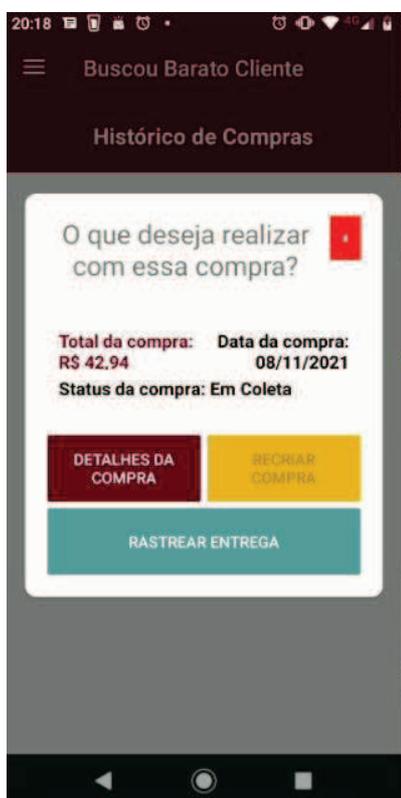
A opção de Recriar compra e Rastrear entrega ficam disponíveis dependendo do status atual da compra, se esta estiver com o status Entregue, a opção de Recriar compra fica habilitada e se o status da compra atualmente for o Em Coleta, a opção Rastrear Compra pode ser utilizada.

A opção de Recriar compra permite o usuário adicionar todos os produtos da compra anterior para o carrinho de compras, para realizar esse procedimento basta tocar no botão Recriar Compra no *pop-up* das opções.

Figura 25 - Tela de histórico de compras



Figura 26 - Opções da compra no histórico de compras



Tela de detalhes da compra

A tela de detalhes mostra os produtos e lojas da compra realizada anteriormente, além da quantidade de itens e valor total. Representada na figura 27, possui também a opção de Recriar compra através dessa tela, o botão é exibido somente quando a compra está com o Status Entregue.

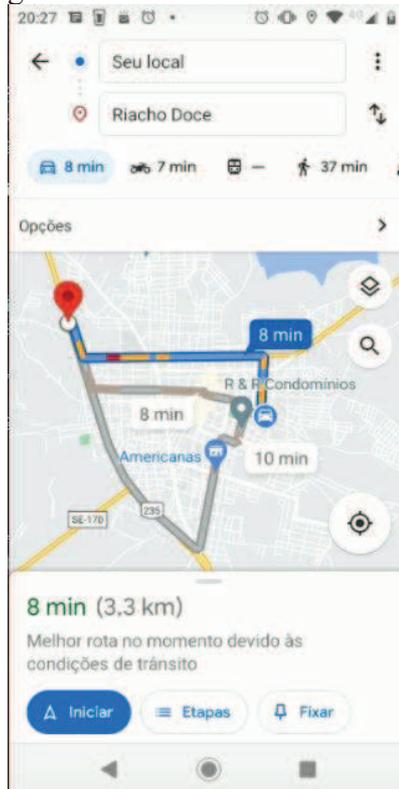
Figura 27 - Tela de detalhes da compra



Tela de rastrear entrega

É possível rastrear a compra se o status da mesma estiver como Em coleta. Ao tocar no botão Rastrear compra das opções na tela de histórico o aplicativo abre o Google Maps e mostra a distância da última parada de coleta do entregador para o endereço do cliente.

Figura 28 - Tela de rastrear entrega



4.2. Buscou Barato Entregador

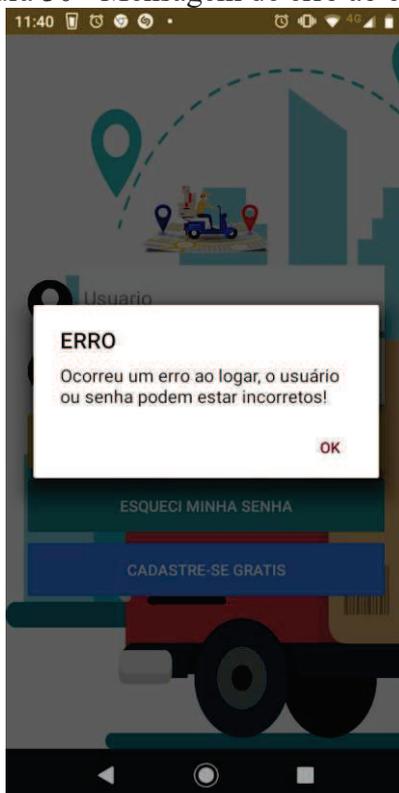
Tela inicial

A tela inicial da aplicação do entregador é a tela de autenticação, se o usuário já efetuou o login quando abrir o aplicativo futuramente ele é redirecionado automaticamente para a tela de paradas nas lojas. Na tela de login assim como a autenticação do aplicativo do cliente necessita dos dados de usuário e senha e tocar no botão Entrar para efetuar a autenticação. Possui o botão de Cadastre-se grátis para realizar o cadastro na aplicação. Essa tela está representada na figura 29. O usuário ou senha pode estar incorreto e a aplicação vai exibir uma mensagem de erro exibida na figura 30.

Figura 29 - Tela inicial de login do aplicativo



Figura 30 - Mensagem de erro ao entrar



Tela de cadastro de usuário

Para utilizar a aplicação o entregador deve possuir uma conta de usuário cadastrada. Durante o cadastro alguns dados são solicitados. Após a realização do cadastro é possível utilizar a aplicação. Essa tela é representada na figura 31.

Figura 31 - Tela de cadastro de usuário

The image displays two side-by-side screenshots of a mobile application's user registration screen. The left screenshot shows the 'Informações da Conta' (Account Information) form. It features a header with a cityscape and a person icon. Below the header are input fields for 'Nome Completo', 'Usuario', 'Senha', 'Confirmação de Senha', 'Telefone', 'Email', and 'Data de Nascimento'. At the bottom, there are radio buttons for 'Masculino' (selected) and 'Feminino'. The right screenshot shows the 'Endereço' (Address) form. It has a header 'Confirmação de Senha' (partially visible) and input fields for 'Telefone', 'Email', 'Data de Nascimento', and gender selection. Below these are fields for 'Logradouro', 'CEP' (with a 'Nº' field), 'Bairro', 'UF', and 'Cidade'. A 'CRIAR CONTA' button is at the bottom.

Menu lateral

No menu lateral contém as funcionalidades principais do aplicativo, ele possui as opções de Início que abre a tela de paradas por loja, Compra já coletadas que abre a tela de compras já coletadas, configurações para mudança de senha e a opção sair para sair e fechar o aplicativo. Representado na figura 32.

Figura 32 - Menu lateral do aplicativo



Tela de paradas por loja

Essa tela mostra a lista de lojas nas quais o entregador possui compras dos clientes para coletar. Essa tela está representada na figura 33, e nela é exibido o nome da loja e quantas compras é necessário recolher naquela loja. Ao tocar no item que equivale a loja, o usuário é exibido o *pop-up* que está representado na figura 34. Nele possui as opções de Visualizar compras que abre a tela de compras por loja, e a opção de Ver no Mapa que abre a tela rastrear endereço da loja que abre no Google Maps a localização da loja e traça o caminho da localização atual até o endereço da loja.

Figura 33 - Tela de paradas por loja

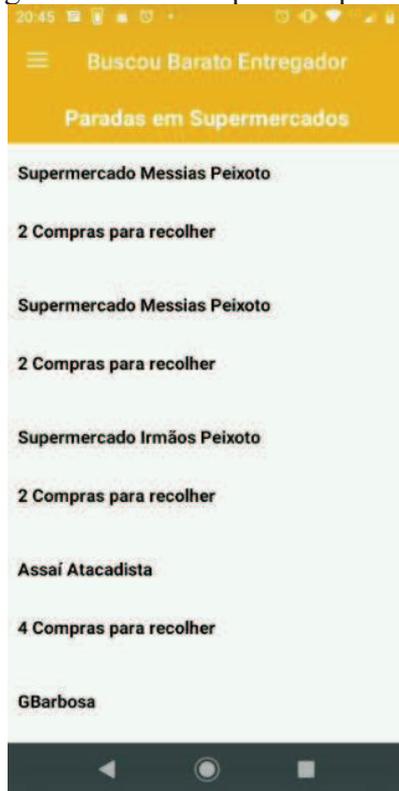
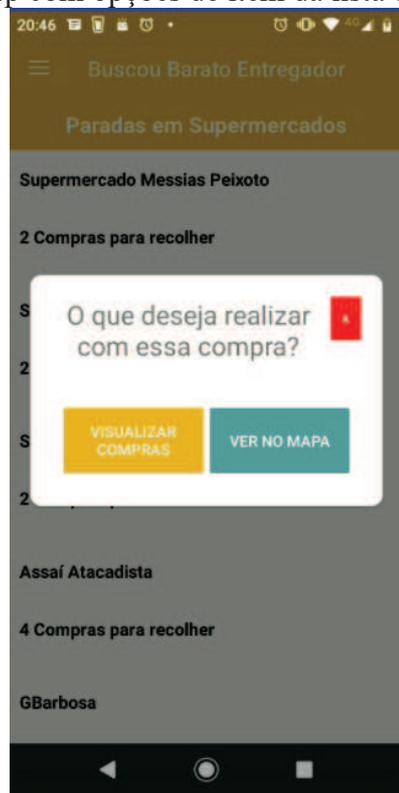


Figura 34 - pop-up com opções de item da lista de paradas por loja



Tela de compras por loja

Essa tela exibe a lista de compras para recolher na loja escolhida na tela anterior. Nela pode ser visto em cada item da lista, o nome do cliente, a data da compra e o valor total da compra. Ao tocar em qualquer item da lista o aplicativo abre a tela de Detalhe da compra. A tela de compras por loja é representada pela figura 35.

Figura 35 - Tela de compras por loja



Tela de detalhe da compra

Essa tela está representada na figura 36 e tem como funcionalidade principal concluir o recebimento dos produtos da compra na loja indicada. É exibido as informações da compra como nome do cliente, data da compra, o nome da loja e a lista de produtos da compra na loja. Para concluir o recebimento o usuário deve tocar no botão Concluir Recebimento. Lembrando que a compra ainda pode não ter sido recolhida totalmente pois o cliente pode comprar produtos de várias lojas na mesma compra.

Figura 36 - Tela de detalhe da compra



Tela de compras já coletadas

Essa tela exibe a lista das compras que já foram recolhidas em todas as lojas e podem ser entregues para o cliente. Possui os dados de nome do cliente, endereço da entrega e a data da compra. A representação dessa tela é a figura 34. Ao tocar no item da lista é aberto um *pop-up* com as opções da compra já recolhida. Esse *pop-up* é representado na figura 37 e possui algumas opções para realizar com a compra. O botão de detalhes, abre a tela de detalhe da compra já coletada. Já o botão de Ver no mapa a tela de rastrear endereço do cliente no google maps e traça a rota da localização atual até o endereço de entrega. Por fim o botão finalizar conclui a entrega e atualiza o status da compra para Entregue.

Figura 37 - Tela de compras já coletadas

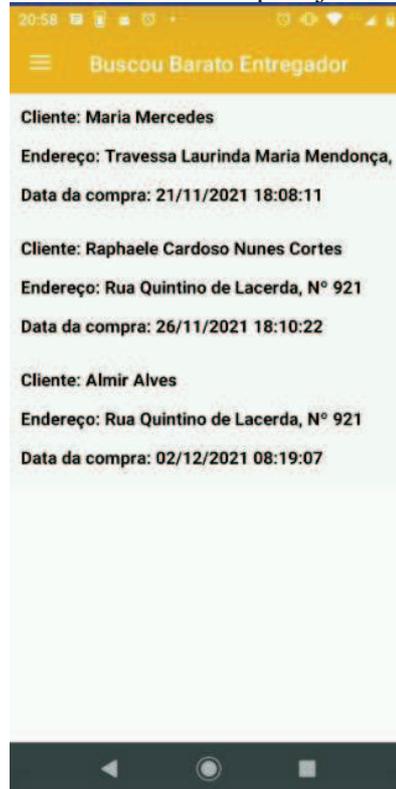
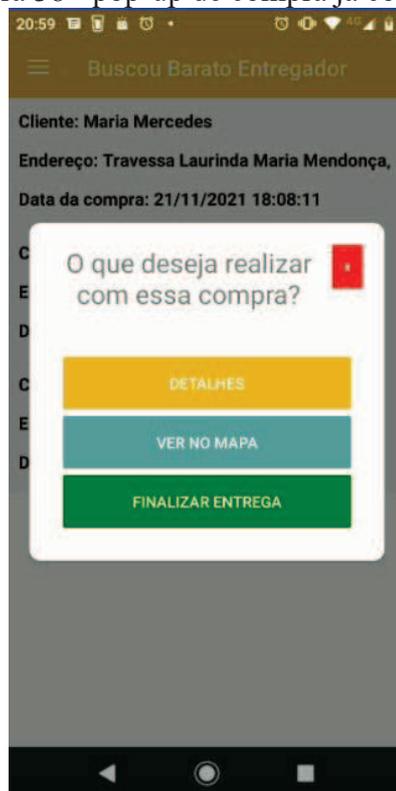


Figura 38 - pop-up de compra já coletada



Tela de detalhe de compra já coletada

Os detalhes da compra já recolhida nas lojas podem ser visualizados nessa tela, exibe as informações de nome do cliente, data da compra e lista de produtos da compra. O botão de finalizar entrega conclui a entrega e atualiza o status da compra para Entregue sendo o último passo de todo o processo do sistema. Essa tela é representada na figura 39.

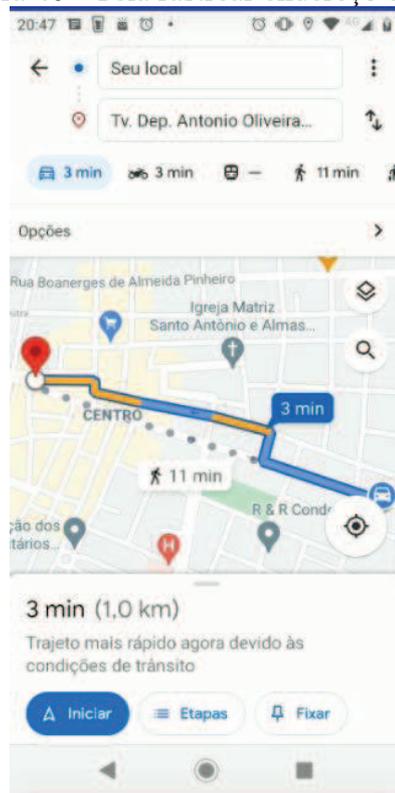
Figura 39 - Tela de detalhamento de compra já coletada



Tela rastrear endereço da loja

Abre o Google Maps com o endereço da loja como destino e traça a rota da localização atual até o endereço da loja.

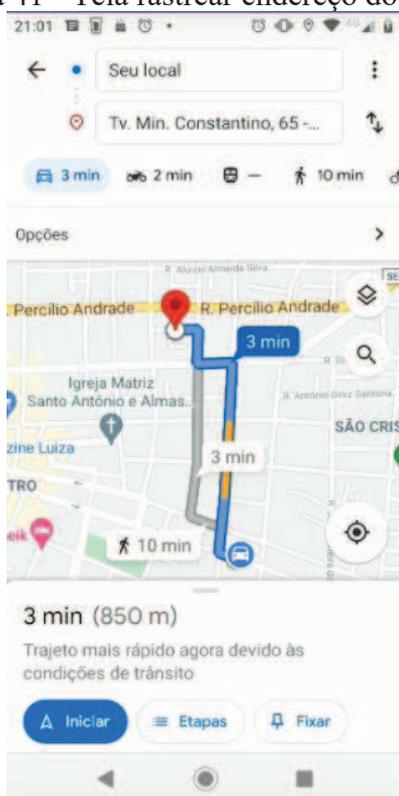
Figura 40 - Tela rastrear endereço da loja



Tela rastrear endereço do cliente

Abre o Google Maps com o endereço para entrega da compra como destino e traça a rota da localização atual até o endereço de entrega.

Figura 41 - Tela rastrear endereço do cliente



4.3 Buscou Barato Lojista

Tela inicial

A aplicação do lojista possui, assim como a aplicação do entregador a tela de login como tela inicial, se o usuário já efetuou o login ao abrir o aplicativo é redirecionado automaticamente para a tela de compras para coletar. Na tela de login assim como a autenticação do aplicativo do cliente e do entregador necessita dos dados de usuário e senha e tocar no botão Entrar para efetuar a autenticação. O botão Cadastre-se grátis também está disponível para efetuar o cadastro do usuário do lojista. A figura 42 representa a tela inicial de autenticação.

Figura 42 - Tela inicial de autenticação



Tela de Cadastro

Assim como nas outras aplicações é necessário possuir uma conta de usuário cadastrada. Durante o cadastro alguns dados são solicitados, inclusive a loja na qual o usuário irá coletar as compras. Após a realização do cadastro é possível utilizar a aplicação. Essa tela é representada na figura 43.

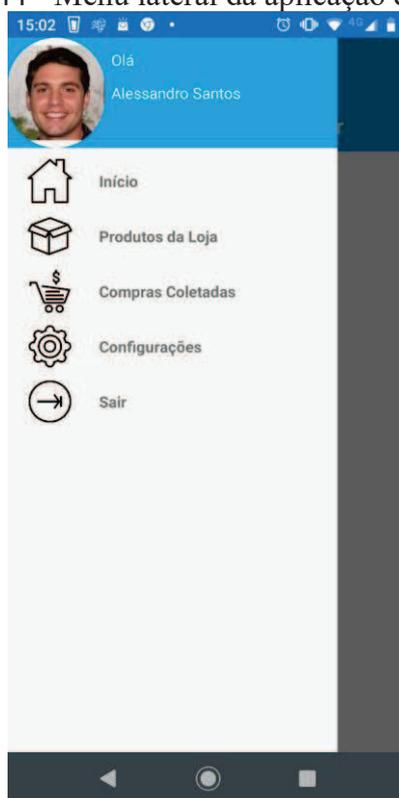
Figura 43 - Tela de cadastro de usuário

The image displays two side-by-side screenshots of a mobile application's user registration screen. The left screenshot, titled "Informações da Conta", features a blue header with a user icon and a list of input fields: "Nome Completo", "Usuario", "Senha", "Confirmação de Senha", "Telefone", "Email", and "Data de Nascimento". Below these fields are radio buttons for "Masculino" (selected) and "Feminino". The right screenshot, titled "Endereço", shows fields for "Logradouro", "CEP", "Nº", "Bairro", "UF", and "Cidade". A green button labeled "CRIAR CONTA" is positioned at the bottom of the right screenshot. Both screenshots show a standard Android status bar at the top with the time 21:09 and 21:10 respectively.

Menu Lateral

No menu lateral estão disponíveis as opções das principais funcionalidades do aplicativo. Contém as opções de Início que abre a tela de Compras para coletar, Produtos da loja que abre a tela de Produtos cadastrados na loja, Compras coletadas que abre a tela de Compras já coletadas, Configurações que abre a tela de Mudar senha e Sair para sair e fechar aplicação. Representado na figura 44.

Figura 44 - Menu lateral da aplicação do lojista



Compras para coletar

Essa é a tela inicial do aplicativo se o usuário já estiver autenticado, ela exibe a lista de compras para coletar na loja. Nela pode ser visto em cada item da lista, o nome do cliente, o telefone do cliente para eventuais contatos com o lojista sobre os produtos comprados na loja e a data da compra. Ao tocar em qualquer item da lista o aplicativo abre a tela de Detalhamento de compra para coletar. A tela de Compras para coletar é representada pela figura 45.

Figura 45 - Tela de compras a coletar



Detalhamento de compra para coletar

O detalhamento da compra exibe a lista de produtos para auxiliar na separação dos itens da compra na loja. Outros dados presentes nessa tela são o nome do cliente, data da compra e telefone. Após o lojista concluir a coleta dos produtos ele deve tocar no botão Concluir Coleta e estará disponível para recolhimento do entregador. O Detalhamento de compra para coletar é representado na figura 46.

Figura 46 - Tela de Detalhamento de compra para coletar



Produtos cadastrados na loja

Essa tela exibe todos os produtos cadastrados na loja. É possível verificar em cada item a imagem do produto e o preço unitário do mesmo. É aberto um *pop-up* ao tocar em qualquer item da lista com as opções do que fazer com aquele produto. As funcionalidades permitidas são de Editar Produto que abre a tela de Alteração de produto, Alterar Estoque que exibe o *pop-up* para alteração da quantidade de estoque armazenado daquele produto e Remover Produto. O *pop-up* das opções está representado na figura 48, e a tela da lista de produtos cadastrados na loja representada na figura 47.

Figura 47 - Tela de lista de produtos cadastrados na loja



Figura 48 - Opções de produto da lista



Adição de produtos na loja

Aqui é possível adicionar produtos na loja. O usuário deve informar o nome do produto no campo e após digitar 4 caracteres uma lista de sugestão vai surgir abaixo do campo e o lojista deve escolher um deles. Não é possível cadastrar um produto que ainda não exista na base de dados para não haver redundâncias ou erros de cadastro, o usuário só consegue adicionar produtos já existentes na base através do campo com a lista de sugestão. Após escolha do produto na lista de sugestão a imagem do produto aparece no campo acima e o lojista deve informar o preço unitário desse produto assim como sua quantidade em estoque. A tela de cadastro de produto está representada na figura 49.

Figura 49 - Tela de lista de Adição de produtos na loja

A imagem mostra a interface de usuário para o cadastro de um produto. O cabeçalho azul contém o ícone de uma seta para trás, o texto 'Buscou Barato Lojista' e 'Cadastro de Produto' com um ícone de caixa. Abaixo, há três campos de entrada: 'Nome do produto', 'Preço do Produto' e 'Quantidade'. O campo 'Quantidade' possui botões de menos e mais. Na base do formulário, há um botão verde com o texto 'ADICIONAR PRODUTO NA LOJA'. O rodapé do aplicativo é visível na parte inferior.

Alteração de produtos

Muito semelhante a tela de adição a tela de alteração já vem com os dados preenchidos do produto escolhido na tela anterior. O usuário deve informar os dados que deseja alterar e tocar no botão Editar produto da loja. Está representada na figura 50.

Figura 50 - Tela de Alteração de produtos



Alterar estoque de produto

Na alteração de estoque do produto da loja o usuário pode alterar a quantidade de estoque e tocar no botão Editar estoque. Esse *pop-up* está representado na figura 51.

Figura 51 - Alteração de estoque de produto



Pesquisa de produtos

Aqui é possível pesquisar os produtos da loja, para chegar nessa tela basta tocar no texto pesquisar no canto superior direito da tela de produtos cadastrados da loja. Para realizar a pesquisa basta informar o nome do produto. Com mais de quatro caracteres digitados o aplicativo traz sugestões de produtos da loja para pesquisa, para finalizar a busca basta tocar no botão Pesquisar. Esse *pop-up* está representado na figura 52.

Figura 52 - Pesquisa de produtos da loja



Mudar Senha

A funcionalidade de mudança de senha é mesma dos aplicativos de cliente e entregador. A tela de mudança de senha representada na figura 53 possui validações, de análise de senha atual, igualdade entre a nova senha e confirmação de senha e a nova senha deve estar no padrão do sistema. Lembrando que o padrão exige que a senha deve conter no mínimo 8 caracteres, e pelo menos, uma letra maiúscula, uma letra minúscula, um caractere especial e um número. Qualquer uma das senhas informadas que não estejam em conformidade gera uma mensagem de erro assim como nas aplicações anteriores. As mensagens podem ser vistas na figura 17.

Figura 53 - Tela para alterar senha



The image shows a mobile application interface for changing a password. At the top, there is a blue header bar with a white hamburger menu icon on the left and the text 'Buscou Barato Lojista' on the right. Below the header, the title 'Mudar Senha' is centered. There are three text input fields stacked vertically, labeled 'Senha Atual', 'Nova Senha', and 'Confirmação de Senha'. Below these fields is a prominent green button with the white text 'TROCAR SENHA'. The bottom of the screen shows the standard Android navigation bar with back, home, and recent apps icons.

Compras já coletadas

Essa tela exibe a lista das compras que já foram coletadas na loja e estão à espera do entregador para recolhimento. Possui os dados de nome do cliente, telefone e a data da compra. A representação dessa tela é a figura 54. Ao tocar no item da lista abre a tela de detalhe da compra já coletada.

Figura 54 - Tela de lista de compras já coletadas



Detalhamento de compra já coletada

Os detalhes da compra já coletada na loja podem ser visualizados nessa tela, exibe as informações de nome do cliente, data da compra e telefone, além da lista de produtos da compra. A compra nesse momento aguarda o recolhimento do entregador e após todo o recolhimento nas lojas onde o cliente comprou produtos o entregador deverá finalizar a entrega no endereço do cliente. A tela de detalhamento está representada na figura 55.

Figura 55 - Tela de detalhamento de compra já coletada



5 CONCLUSÕES E TRABALHOS FUTUROS

Muitas aplicações de *delivery* foram desenvolvidas nos últimos anos, com o principal objetivo de facilitar a vida de seus usuários ao realizar suas compras. Com o intuito de auxiliar a busca pelos produtos mais baratos da região, esse trabalho tem como objetivo a criação de um sistema de *delivery*.

Neste trabalho, foi proposto um sistema com três aplicações, uma para cada tipo de usuário, que oferece aos clientes a busca pelo menor preço de produtos em lojas próximas do comércio varejista da região, utilizando geolocalização, e a entrega integral das compras efetuadas por entregadores cadastrados no sistema.

O desenvolvimento foi motivador, pois o trabalho contribui para a comunidade ao gerar economia para os clientes que poderão efetuar as compras mais baratas, e sem sair da sua residência, gerando um novo meio de vendas e concorrência de preços entre as lojas, e por fim, emprego para entregadores que poderão trabalhar para recolher e entregar as compras dos clientes.

Para futuras evoluções, visando dar continuidade ao projeto apresentado, destacam-se as principais melhorias a serem realizadas: Realização de pagamentos utilizando outros meios como cartão de crédito; funcionalidade para esqueceu a senha; cadastro de conta bancária no app do entregador para o mesmo receber os pagamentos pelas entregas realizadas; gerar documento com *qr code* no aplicativo do entregador para recolhimento da compra do cliente na loja; leitor do *qr code* do entregador no aplicativo do lojista para liberar a compra; notificações em segundo plano da aplicação para mudança de status da entrega para os clientes, entregadores e lojistas; criação de módulo web para gerar relatórios das vendas do aplicativo, para analisar o cadastro e aceitação de novos entregadores com o envio de documentos de carteira de motorista, certidão de antecedentes criminais e comprovante de residência, cadastro de novas lojas e redes de lojas e produtos, atualmente o cadastro é realizado diretamente no banco de dados.

REFERÊNCIAS

ASSOCIAÇÃO PAULISTA DE SUPERMERCADOS. **APAS revela pesquisa inédita sobre tendências do consumidor e dados do setor supermercadista durante Feira e Congresso.** Disponível em: <<http://feiraapas.com.br/wp-content/uploads/2016/05/COLETIVA-Pesquisa-APAS-Nielsen-Kantar.pdf>>. Acesso em: 26 set. 2016

BOHRER, F. J. **SERVIÇO DE GEOLOCALIZAÇÃO PARA PLATAFORMA ANDROID.** Centro Universitário UNIVATES, 2011.

CANALLE, A. **Empregando tecnologia Java, Android e Geoprocessamento em aplicativos móveis.** Universidade Tecnológica Federal do Paraná, 2011.

COMTECH. **Pesquisa Anual de Vendas de Dispositivos Móveis por Sistema Operacional.** Disponível em: <<http://www.kantarworldpanel.com/global/smartphone-os-market-share/>>. Acesso em: 20 out. 2021.

CORAZZA, R. M. F. **Desenvolvimento de um sistema móvel para o rastreamento dos processos produtivos de pequenas propriedades rurais.** Centro Universitário Eurípides De Marília – UNIVEM, 2014.

DJUKNIC, G. M.; RICHTON, R. E. **Geolocation and assisted GPS.** Computer, v. 34, n. 2, p. 123–125, 2001.

ERL, T. **SOA - Princípios de Design de Serviços.** São Paulo: Pearson, 2009.

GUINARD, D.; TRIFA, V.; KARNOUSKOS, S.; SPIESS, P.; SAVIO, D. **Interacting with the SOA-based internet of things: Discovery, query, selection, and on-demand provisioning of web services.** IEEE Transactions on Services Computing. *Anais...*2010

HOENTSCH, S. C. P. **Socialnetlab : Serviços De Localização Através Da Api De Geolocalização Do Html5 E Do Web.** Universidade Federal de Sergipe, 2013.

KENNEDY, S.; STEWART, R.; JACOB, P.; MOLLOY, O. **StoRHm: A protocol adapter for mapping SOAP based Web Services to RESTful HTTP format.** Electronic Commerce Research, v. 11, n. 3, p. 245–269, 2011.

LI, H. **RESTful Web service frameworks in Java.** IEEE International Conference on Signal Processing, Communications and Computing, ICSPCC 2011. *Anais...*2011

MARY, T. C.; VINOSHRI, M. **A Survey Paper on Soap , Wsdl , Uddi Approach for Web Service Accesses.** International Journal of Advanced Research in Information and Communication Engineering, v. 1, n. 2, p. 1–7, 2013.

MEIRELLES, F. S. **32ª Pesquisa Anual do Uso de TI,** FGV-EAESP-CIA, 2021. Disponível em: < <https://eaesp.fgv.br/sites/eaesp.fgv.br/files/u68/fgvcia2021pesti-relatorio.pdf> >. Acesso em: 13 out. 2021.

MELO, M. C. DE; DAVID, M.; RIBEIRO, D. A.; LUCAS, C. A. **FASTLINE : Aplicativo mobile para lanchonetes**. Revista Eletrônica de Sistemas de Informação e Gestão Tecnológica. p. 14, 2015.

SILVA, M. R. G. **O CRESCIMENTO DAS EMPRESAS DE DELIVERY NO CONTEXTO DA PANDEMIA**. Faculdade Federal de Minas Gerais, Centro de Pós-Graduação e Pesquisas em Administração, Belo Horizonte, 2021.

NIGRI, I. J. **DELIVERY X BALCÃO – UM ESTUDO DE CASO EM UMA REDE DE FAST FOOD**. Universidade Federal do Rio de Janeiro, Centro de Ciências Jurídicas e Econômicas, Rio de Janeiro, 2019.

MELO, C. F. A. **BOM DELIVERY: UM APLICATIVO DE DELIVERY GENÉRICO PARA CIDADES DO INTERIOR**. Universidade Federal de Campina Grande, Centro de Engenharia Elétrica e Informática, Campina Grande, 2019.

MONTEIRO, B. D. S. **Ambiente de Aprendizado Ubíquo Youubi: Design e Avaliação**. Universidade Federal de Pernambuco, 2015.

MORO, T. D.; DORNELES, C. F.; REBONATTO, M. T. **Web services WS- * versus Web Services REST**. REIC - Revista de Iniciação Científica, v. 11, n. 1, p. 36–51, 2011.

O'BRIEN, L.; MERSON, P.; BASS, L. **Quality Attributes for Service-Oriented Architectures**. International Workshop on Systems Development in SOA Environments (SDSOA'07: ICSE Workshops 2007). **Anais...2007**

CONSTANTIN, N. O., Robert-Gyula, M., e Cornelia, N. M. **Comparison of hybrid cross-platform mobile applications with native cross-platform applications**. Journal Of Computer Science And Control Systems, v.9, p. 24-27, 2016.

OLIVEIRA, F. A. DE. **Geolocalização , redes sociais e dispositivos móveis : proposta de sistema gráfico**. Universidade do Estado do Rio de Janeiro, 2013.

PAPAZOGLU, M. P.; HEUVEL, W. J. VAN DEN. **Service oriented architectures: Approaches, technologies and research issues**. VLDB Journal, v. 16, n. 3, p. 389–415, 2007.

ROXIN, A.; GABER, J.; WACK, M.; NAIT-SIDI-MOH, A. **Survey of wireless geolocation techniques**. GLOBECOM - IEEE Global Telecommunications Conference, 2007.

SAUDATE, A. **SOA Aplicado Integrando com web services e além**. São Paulo: Casa do Código, 2014.

BOAGLIO, F. **Spring Boot. Acelere o desenvolvimento de microserviços**. São Paulo: Casa do Código, 2017.

SAUDATE, A. **REST Construa API's inteligentes de maneira simples**. São Paulo: Casa do Código, 2016.

TANNER, A. **Xamarin forms: For starters**, 2016.

LOPES, S. **A web mobile: programe para um mundo de dispositivos**. São Paulo: Casa do Código, 2013.

BOCARD, T. **Quanto custa um aplicativo?** 2021. Disponível em <<https://usemobile.com.br/quanto-custa-um-aplicativo/>>. Acesso em: 29 out. 2021.

SOUZA, R.; NAKAMURA, L.; PRADO, P.; SANTANA, M. J. **Avaliação de desempenho entre Web Services SOAP e RESTful com o uso do Apache CXF**. Researchgate.Net, n. Icmc, p. 1–4, 2012.

JOBE, W. **Native apps vs. mobile web apps**. International Journal of Interactive Mobile Technologies (iJIM), 2013.

SILVA, M; SANTOS, M. **Os Paradigmas de Desenvolvimento de Aplicativos para Aparelhos Celulares**. Departamento da Computação: Universidade Federal de São Carlos, 2014.

PAIVA, F. **79,3% dos brasileiros têm celular, informa IBGE**, 2020. Disponível em: <<https://teletime.com.br/29/04/2020/793-dos-brasileiros-tem-celular-informa-ibge/>>. Acesso em: 15 out. 2021.

ROMMI, M. **5 Advantages and Disadvantages of Web Application | Drawbacks & Benefits of Web Application**, 2021 Disponível em: <<https://www.hitechwhizz.com/2021/04/5-advantages-and-disadvantages-drawbacks-benefits-of-web-application.html>> Acesso em: 29 out. 2021.

IFOOD iFood – Delivery de Comida, versão 1.23.0, iFood, 2021. Disponível em <<https://www.ifood.com.br/>>. Acesso em: 28 out. 2021.

EXTRA Extra, versão 1.0, Cnova, 2021. Disponível em <<http://www.deliveryextra.com.br/>>. Acesso em: 27 out. 2021.

TECHOPEDIA: **Educating IT Professionals To Make Smarter Decisions. CROSS-PLATFORM DEVELOPMENT**, 2020. Disponível em: <<https://www.techopedia.com/definition/30026/cross-platform-development>> Acesso em: 23 out. de 2021.

XAMARIN. **O que é o Xamarin?** 2021. Disponível em: < <https://docs.microsoft.com/pt-br/xamarin/get-started/what-is-xamarin>> Acesso em: 30 out. de 2021.

UberEats. Disponível em: <<https://merchants.ubereats.com/br/pt-br/>>. Acessado em 03 nov. De 2021.

MOVIMENTO ECONÔMICO. **A sergipana Quero Delivery foca no interior para crescer.** 2021. Disponível em: <<https://movimentoeconomico.com.br/empresas/2021/09/23/a-sergipana-quero-delivery-foca-no-interior-para-crescer/>> Acesso em: 30 out. de 2021.

REDE FOOD SERVICE. **Quero Delivery: o maior app do Nordeste que cresceu 400% durante a pandemia de Covid-19.** 2021. Disponível em: <<https://redefoodservice.com.br/2021/10/quero-delivery-o-maior-app-do-nordeste-que-cresceu-400-durante-a-pandemia-de-covid-19/>> Acesso em: 30 out. de 2021.

O Impacto da COVID-19 nas vendas: Disponível em:
<<https://www.sebrae.com.br/Sebrae/Portal%20Sebrae/Anexos/Nielsen%20-%20Impactos%20da%20COVID-19%20nas%20vendas%20de%20produtos%20de%20consumo%20de%20giro%20rA%CC%83%C2%A1pido%20no%20Brasil%20e%20ao%20redor%20do%20mundo.pdf.pdf>>:
Acesso em 01 de nov. 2021.