

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DO AMAZONAS
CAMPUS MANAUS CENTRO**

Victor Duarte Nascimento

**SISTEMA MOBILE DE GERENCIAMENTO DE
ENTREGAS PARA PEQUENAS EMPRESAS**

Manaus, Amazonas – Brasil
2022

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E
TECNOLOGIA DO AMAZONAS
CAMPUS MANAUS CENTRO**

Victor Duarte Nascimento

**SISTEMA MOBILE DE GERENCIAMENTO DE
ENTREGAS PARA PEQUENAS EMPRESAS**

“Trabalho de Conclusão de Curso
apresentado à banca examinadora
Curso Superior de Tecnologia em
Análise e Desenvolvimento de
Sistema do Instituto Federal de
Educação, Ciências e Tecnologia do
Amazonas – IFAM Campus Manaus -
Centro, como requisito para o
cumprimento da disciplina TCC II –
Desenvolvimento de Software”

**Prof. MSc. Emmerson Santa
Rita da Silva**

Dezembro /2022
Manaus, Am

Biblioteca do IFAM – Campus Manaus Centro

N244s Nascimento, Victor Duarte
Sistema mobile de gerenciamento de entregas para pequenas empresas /
Victor Duarte do Nascimento. – Manaus, 2022.
67 p. : il. color.

Trabalho de Conclusão de Curso (Tecnologia em Análise e
Desenvolvimento de Sistema) – Instituto Federal de Educação, Ciência e
Tecnologia do Amazonas, *Campus* Manaus Centro, 2022.

Orientador: Prof. Me. Emmerson Santa Rita da Silva.

1. Desenvolvimento de sistema. 2. Gestão de monitoramento. 3.
Logística. 4. E-commerce. 5. Pequenas empresas. I. Silva, Emmerson
Santa Rita da. (Orient.) II. Instituto Federal de Educação, Ciência e
Tecnologia do Amazonas III. Título.

CDD 005.3

FOLHA DE APROVAÇÃO

VICTOR DUARTE NASCIMENTO

Esta monografia de Conclusão de Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Manaus Centro, foi julgada e Aprovada pela Banca Examinadora:

Prof.º MSc. EMMERSON SANTA RITA DA SILVA
ORIENTADOR

Prof.º MSc. ANTONIO FERREIRA DOS SANTOS JUNIOR
EXAMINADOR 1

Prof.º MSc. SERGIO AUGUSTO COELHO BEZERRA
EXAMINADOR 2

Dezembro /2022
Manaus, Am

DEDICATORIA

A Deus, que me deu forças e me deu sabedoria para executar este trabalho. Minha mãe e meu pai por sempre me apoiarem e não medirem esforços nesse caminho para chegar até aqui.

RESUMO

A logística tem um papel fundamental no cotidiano, seja em meios empresariais ou seja no abastecimento de recursos. Desde os primórdios do seu surgimento como meios de abastecimento à soldados em meio a guerra. Com o passar das guerras tornou-se uma área extremamente importante principalmente em pequenas empresas onde os recursos são menores uma logística bem aplicada otimiza gastos e acelera atividades. Objetiva-se, neste trabalho o desenvolvimento de uma ferramenta de auxílio à profissionais da área, principalmente pequenas empresas, unindo o uso de uma tecnologia mais acessível como os dispositivos mobile. Este trabalho baseou-se em um modelo de pesquisa bibliográfica de natureza aplicada, de maneira a auxiliar na solução de problemas específicos. Através de uma análise do funcionamento logístico foi desenvolvido uma aplicação com intuito de contribuir para a confiabilidade e organização das atividades. Através da utilização da ferramenta obter uma noção de gerenciamento de uma parte das atividades logísticas, justamente o transporte das mercadorias. Para o desenvolvimento da ferramenta proposta, utilizou-se tecnologias acessíveis com o intuito de auxiliar no gerenciamento das atividades de abastecimento e entregas.

Palavras-Chave: Logística; Pequenas Empresas; *E-commerce*; Gestão e Monitoramento; Aplicativo *Mobile*.

ABSTRACT

Logistics plays a fundamental role in daily life, whether in business environments or in the supply of resources. Since the beginning of its emergence as a means of supply to soldiers in the midst of war. With the passing of the wars it became an extremely important area, especially in small companies where resources are smaller. A well applied logistics optimizes expenses and accelerates activities. The objective of this work is to develop a tool to help professionals in the area, especially small businesses, combining the use of a more accessible technology such as mobile devices. This work was based on a model of bibliographical research of an applied nature, in order to assist in the solution of specific problems. Through an analysis of the logistics operation was developed an application in order to contribute to the reliability and organization of activities. Through the use of the tool to obtain a notion of management of a part of the logistic activities, precisely the transportation of goods. For the development of the proposed tool, it was used accessible technologies in order to assist in the management of supply activities and deliveries.

Keywords: Logistics. Small Businesses. E-commerce. Management and Monitoring. Mobile Application

LISTA DE FIGURAS

Figura 1 - Categoria das atividades logísticas	19
Figura 2 - Taxa de sobrevivência de pequenas empresas	20
Figura 3 - Usuário realiza pedido por um aplicativo	23
Figura 4 - IDE executando teste de performance no emulador	24
Figura 5 - Console do Firebase administrando um banco de dados	25
Figura 6 - Popularidade da linguagem Kotlin	27
Figura 7 - Ciclo de vida do framework SCRUM	28
Figura 8 - Dashboard do Sistema FW Pizza	29
Figura 9 - Sistema SHIPPING utilizando a função de rastreamento do entregador	30
Figura 10 - Diagrama de casos de uso	34
Figura 11 - Diagrama de Classes	45
Figura 12 - Distribuição dos documentos dentro do Banco de Dados	46
Figura 13 - Módulos de Acesso do aplicativo	48
Figura 14 - Menu de navegação	49
Figura 15 - Formulários de cadastros	49
Figura 16 - Formulários de cadastro e agendamento	50
Figura 17 - Visualização e lista de clientes	51
Figura 18 - Cadastro de encomenda e Seleção de cliente	51
Figura 19 - Visualização e envio do código de rastreamento para o cliente	52
Figura 20 - Sessão e visualização de entregador	53
Figura 21 - Opções da visualização do entregador	53
Figura 22 - Alerta com ações disponíveis e edição de dados da Empresa	54
Figura 23 - Modulo do Cliente	55
Figura 24 - Tela de Login e Entregas pendentes do Entregador	55
Figura 25 - Telas de trajeto e registro de ocorrência	56
Figura 26 – Arquitetura MVVM	33
Figura 27 - Projeto aplicando a arquitetura MVVM	35
Figura 28 - Dependências necessárias no arquivo Gradle.	36
Figura 29 - Código das data classes utilizadas	36
Figura 30 - Classe EntregadorRepository	37
Figura 31 - Classe EntregadorViewModel	39
Figura 32 - Classe EntregadoresFragment	40
Figura 33 - Classe AcompanharTrajetoActivity	41

LISTA DE TABELAS

Figura 1 - Categoria das atividades logísticas	19
Figura 2 - Taxa de sobrevivência de pequenas empresas	20
Figura 3 - Usuário realiza pedido por um aplicativo	23
Figura 4 - IDE executando teste de performance no emulador	24
Figura 5 - Console do Firebase administrando um banco de dados	25
Figura 6 - Popularidade da linguagem Kotlin	27
Figura 7 - Ciclo de vida do framework SCRUM	28
Figura 8 - Dashboard do Sistema FW Pizza	29
Figura 9 - Sistema SHIPPING utilizando a função de rastreo do entregador	30
Figura 10 - Diagrama de casos de uso	34
Figura 11 - Diagrama de Classes	45
Figura 12 - Distribuição dos documentos dentro do Banco de Dados	46
Figura 13 - Módulos de Acesso do aplicativo	48
Figura 14 - Menu de navegação	49
Figura 15 - Formulários de cadastros	49
Figura 16 - Formulários de cadastro e agendamento	50
Figura 17 - Visualização e lista de clientes	51
Figura 18 - Cadastro de encomenda e Seleção de cliente	51
Figura 19 - Visualização e envio do código de rastreo para o cliente	52
Figura 20 - Sessão e visualização de entregador	53
Figura 21 - Opções da visualização do entregador	53
Figura 22 - Alerta com ações disponíveis e edição de dados da Empresa	54
Figura 23 - Modulo do Cliente	55
Figura 24 - Tela de Login e Entregas pendentes do Entregador	55
Figura 25 - Telas de trajeto e registro de ocorrência	56

LISTA DE ABREVIATURAS E SIGLAS

CNPJ	<i>Cadastro Nacional de Pessoas Jurídicas</i>
CPF	<i>Cadastro de Pessoas Físicas</i>
ERP	<i>Enterprise Resource Planning</i>
IDE	<i>Integrated Development Environment</i>
SEBRAE	<i>Serviço Brasileiro de Apoio às Micro e Pequenas Empresas</i>
TCC	<i>Trabalho de Conclusão de Curso</i>

SUMÁRIO

1	INTRODUÇÃO	12
1.1.	PROBLEMATIZAÇÃO.....	13
1.2.	JUSTIFICATIVA	14
1.3.	OBJETIVOS	14
1.3.1.	Objetivo geral	14
1.3.2.	Objetivos específicos	15
1.4.	METODOLOGIA	15
1.5.	ORGANIZAÇÃO DA MONOGRAFIA	17
2	FUNDAMENTAÇÃO TEÓRICA.....	18
2.1.	INTRODUÇÃO	18
2.2.	LOGÍSTICA	18
2.3.	PEQUENAS EMPRESAS	19
2.4.	E-COMMERCE	21
2.5.	GESTÃO E MONITORAMENTO.....	21
2.6.	APLICATIVO MOBILE.....	22
2.7.	FERRAMENTAS DE DESENVOLVIMENTO DO SOFTWARE	23
2.7.1.	<i>Android Studio</i>	23
2.7.2.	<i>Android</i>	24
2.7.3.	<i>Firebase</i>	24
2.7.4.	<i>NoSQL</i>	25
2.7.5.	<i>Kotlin</i>	26
2.7.6.	<i>SCRUM</i>	27
2.8.	TRABALHOS RELACIONADOS	28
2.8.1.	Sistema de Pizzaria FW PIZZA.....	28
2.8.2.	Aplicação Web de Gerenciamento de Entregas de Transportadoras - SHIPPING	29
2.8.3.	TudoEntregue	30
2.8.4.	Comparativo	31
3	PROJETO DO SISTEMA	32
3.1.	INTRODUÇÃO	32
3.2.	MODELAGEM DO SISTEMA.....	32
3.3.	DIAGRAMA DE CASOS DE USO	34
3.4.	DESCRIÇÃO DO CASO DE USO.....	35
3.5.	DIAGRAMA DE CLASSES.....	45
3.6.	MODELO DO BANCO DE DADOS	46
4	IMPLEMENTAÇÃO	47
4.1.	INTRODUÇÃO	47

4.2.	FUNCIONALIDADES DO SISTEMA	47
4.3.	APRESENTAÇÃO DO SISTEMA.....	48
4.3.1.	Configuração, Controle de Acesso e Modulo da Transportadora.....	48
4.3.2.	Modulo do Cliente	54
4.3.3.	Modulo do Entregador	55
4.4.	IMPLEMENTAÇÃO DO SISTEMA	56
4.4.1.	Método de desenvolvimento	56
4.4.2.	Arquitetura do projeto.....	58
4.4.3.	Configurações iniciais	59
4.4.4.	Utilização do MVVM	59
4.4.5.	Buscando a localização com as APIs do <i>Google Maps</i>	63
5	CONSIDERAÇÕES FINAIS E DIRECIONAMENTO PARA TRABALHOS FUTUROS	65
5.1.	INTRODUÇÃO	65
5.2.	CONCLUSÃO	65
5.3.	TRABALHOS FUTUROS	65
6	REFERÊNCIAS.....	66

1 INTRODUÇÃO

Tendo em vista o alto crescimento de adeptos ao *E-commerce* e uma vez que é uma área altamente lucrativa, onde as atividades logísticas tornam-se cruciais para o sucesso e a visibilidade da empresa. Uma atividade importante demanda de uma sistematização do monitoramento e gestão. Entre as atividades importantes, pode-se destacar a logística.

A logística engloba o processo de planejar, gerenciar e monitorar o transporte de mercadorias e está presente em todo negócio. Pesquisas apontam que grande parte dos problemas de lojas virtuais são logísticos relacionados a extravios, demora nas entregas e danos a produtos (RICARDO, 2020). Esses fatores podem afetar de maneira negativa empresas de vários portes, entre elas as pequenas empresas.

Responsável pela maior parte das vagas de empregos no Brasil nos últimos anos, a quantidade de Pequenas Empresas vem crescendo consideravelmente, mesmo diante de uma pandemia, conseguiu se manter ativa e em alta utilizando a internet para se sustentar no mercado. (“Pandemia faz Brasil ter recorde de novos empreendedores”, 2020). Logo, fatores que podem ser determinantes para o sucesso de uma pequena empresa, como um bom planejamento e processo logístico tornam-se muito importantes.

Diante da pandemia de Covid-19 que assolou o mundo todo, Pequenas Empresas perceberam a necessidade de se modernizar, desta forma se digitalizando para um meio mais acessível como a internet. Dependendo da tecnologia em todas as etapas do processo de venda e gerenciamento. Consequentemente, a mudança para o meio digital traz a necessidade de um ótimo processo logístico, que possa ser rentável, seguro e eficaz.

Torna-se indispensável para o processo logístico alguns fatores, sendo eles um dos principais alicerces para um processo Logístico de qualidade e eficaz é a união entre gestão e monitoramento sólidos. A gestão e monitoramento têm como objetivo fornecer um maior controle e organização para a empresa, oferecendo garantias quanto a segurança e comprometimento da empresa com o cliente.

Atrelado a esta necessidade de gestão e monitoramento, devido à alta demanda em serviços de compra e venda online, destacam-se o uso de aplicativos mobile, devido a sua praticidade, levando em consideração que é um objeto de uso diário, principalmente para Pequenas Empresas, em razão de que as mesmas possuem um orçamento limitado em relação a Grandes Empresas.

Diante desta realidade, as tecnologias podem ser importantes para o desenvolvimento de mecanismos ágeis para os processos logísticos de pequenas empresas. Essas tecnologias são capazes de gerenciar e monitorar todo o processo logístico até o cliente, para evitar gastos desnecessários e trazer maior segurança em uma via dupla empresa/cliente.

1.1. PROBLEMATIZAÇÃO

Visto que a logística, é todo o processo de planejamento, implementação e o controle eficiente e eficaz do fluxo e armazenagem de mercadorias, serviços e informações relacionadas do ponto de origem ao ponto de consumo, com o foco sempre às necessidades do cliente (CSCMP, 1991). Tendo esse conhecimento, infere-se que os processos logísticos visam solucionar os problemas de entregas e abastecimentos. Em uma empresa ou negócio online, acaba por ser uma atividade complexa para gerenciar.

Um problema comum que ocorre em negócios online, é geralmente relacionado a esta parte logística, seja ela a entrega de produtos, mercadorias, dificuldade de monitoramento e a falta do gerenciamento desses envios. Problemas esses, que são obstáculos tanto para o cliente fornecedor quanto para o cliente destinatário da entrega.

A sistematização da forma de gerenciar e monitorar os processos logísticos viabiliza a maximização da eficiência na execução dessas operações. Apesar de existirem alternativas para esse problema, ainda sim, são de difícil acesso para pequenas empresas que estão iniciando. Então, como auxiliar estas pequenas empresas a gerenciar esses processos?

1.2. JUSTIFICATIVA

Os sistemas de informação, tem papel fundamental na ligação entre as atividades logísticas à um único processo integrado. De maneira que o uso de tecnologias presentes no mercado se torna um importante elemento como fator diferencial para uma empresa que deseja se tornar competitiva demonstrando um diferencial em seu ramo (FERREIRA E RIBEIRO, 2003).

Desta forma o uso de um mecanismo sistematizado, torna-se uma opção para auxiliar principalmente o pequeno empresário, para o gerenciamento e monitoramento. Esse tipo de sistematização precisa ser online para o gerenciamento da parte logística da empresa. Tendo o controle do que entra e o que sai, monitorando status de entregas, estabelecendo metas e limites.

Todas essas possibilidades surgem a partir de uma aplicação que pode tornar os processos logísticos mais eficientes, oferecendo ainda maior segurança e credibilidade nas entregas.

Portanto, o desenvolvimento de aplicações que possam auxiliar principalmente pequenas empresas, torna-se indispensáveis para que alcance um grau de desenvolvimento das mesmas em larga escala.

1.3. OBJETIVOS

1.3.1. Objetivo geral

Projetar e desenvolver um aplicativo *mobile*, que possibilite o gerenciamento e monitoramento de entregas logísticas no âmbito de pequenas empresas.

1.3.2. Objetivos específicos

- Analisar e evidenciar através da pesquisa bibliográfica, ferramentas semelhantes.
- Identificar principais dificuldades e obstáculos nas atividades logísticas.
- Projetar e implementar uma solução que supra as necessidades do pequeno empresário sobre a logística de entregas.
- Auxiliar na integração dos pequenos negócios ao *e-commerce*.

1.4. METODOLOGIA

Esta pesquisa é de natureza aplicada, como foi classificado por SILVEIRA e CÓRDOVA (2009), pois visa a geração de conhecimentos que serão aplicados na prática, de maneira dirigida à solução de problemas específicos, além de relacionar fatos e preferências locais.

Pode-se classificar quanto a seus objetivos como pesquisa descritiva, pois se adéqua a definição feita por GIL (2008), aponta que as pesquisas descritivas têm como objetivo primordial a descrição das características de determinada população, fenômenos ou relações entre variáveis analisadas, ainda possui uma característica importante onde foca na utilização de técnicas padronizadas para a coleta de dados.

Portanto, neste capítulo o problema será abordado de maneira qualitativa, porque a pesquisa qualitativa não se preocupa com a representação de resultados numéricos e sim aprofundamento das motivações de determinado grupo, sistema ou organização, como categorizado por SILVEIRA e CÓRDOVA (2009), na utilização do método qualitativo, busca-se o porquê das coisas, sem quantificar valores, pois analisa-se dados não métricos podendo se valer de diferentes abordagens.

Como modelo de desenvolvimento do software utilizado o modelo espiral atrelado ao SCRUM onde estará organizada por pequenos ciclos que serão definidos como sprints, visando a entrega de produtos com máximo valor conforme mostrado na Figura 7.

Para o processo de desenvolvimento do sistema, serão utilizadas as seguintes etapas: análise e especificações de requisitos, projeto, implementação, testes e implantação. Na etapa de análise e especificações de requisitos será realizada uma pesquisa sistemático-bibliográfica, em busca de características de sistemas correlatos. Durante a etapa de projeto, será realizada uma análise dos dados e requisitos obtidos na etapa de análises, será projetado o sistema que satisfaça o máximo de requisitos possíveis. No decorrer da etapa de implementação será desenvolvido a aplicação com as funcionalidades projetadas na etapa de projeto. No decurso da etapa de testes, será realizado testes, para verificar a usabilidade e se a aplicação atende o que foi projetado. Já na etapa final, de implantação será implantado em um ambiente real, para inferir-se resultados de sua aplicação prática no ambiente para qual foi desenvolvida.

1.5. ORGANIZAÇÃO DA MONOGRAFIA

Este trabalho, está dividido em 6 capítulos, onde no capítulo 1 encontra-se a Introdução, em que se apresenta uma contextualização do tema abordado, mostra os objetivos a serem alcançados e a justificativa para a proposta do projeto.

O capítulo 2 tem-se a Fundamentação Teórica, trazendo alguns conceitos importantes, para o entendimento e compreensão de como foi desenvolvido o sistema, e as tecnologias utilizadas, além de apresentar trabalhos correlatos que ajudaram no desenvolvimento do trabalho em geral.

O capítulo 3 tem como objetivo apresentar de maneira resumida a proposta de solução para o problema abordado na pesquisa, para melhor entendimento será utilizado a Linguagem Unificada para Modelagem (UML – *Unified Modeling Language*). Serão apresentados diagramas utilizados na modelagem de banco, do sistema e funcionalidades.

O capítulo 4 apresenta a implementação do sistema, como foi implementado e organizado. Mostra ainda algumas partes importantes para sua implementação.

O capítulo 5 disserta as considerações finais, mostrando os resultados que foram obtidos e que podem ser melhorados em trabalhos futuros com relação ao tema abordado.

O capítulo 6 traz as referências utilizadas na construção da monografia, assim como os autores e *links* das páginas consultadas.

2 FUNDAMENTAÇÃO TEÓRICA

2.1. INTRODUÇÃO

De acordo com as pesquisas realizadas visando maior compreensão do tema proposto, a fonte dos materiais utilizados para o desenvolvimento da proposta de solução do presente trabalho foi de fontes bibliográficas tais como livros, artigos, trabalhos acadêmicos e sites de organizações oficiais. Nesta sessão, serão conceituados os tópicos relevantes para a pesquisa realizada e a visão de autor e organização sobre seus temas.

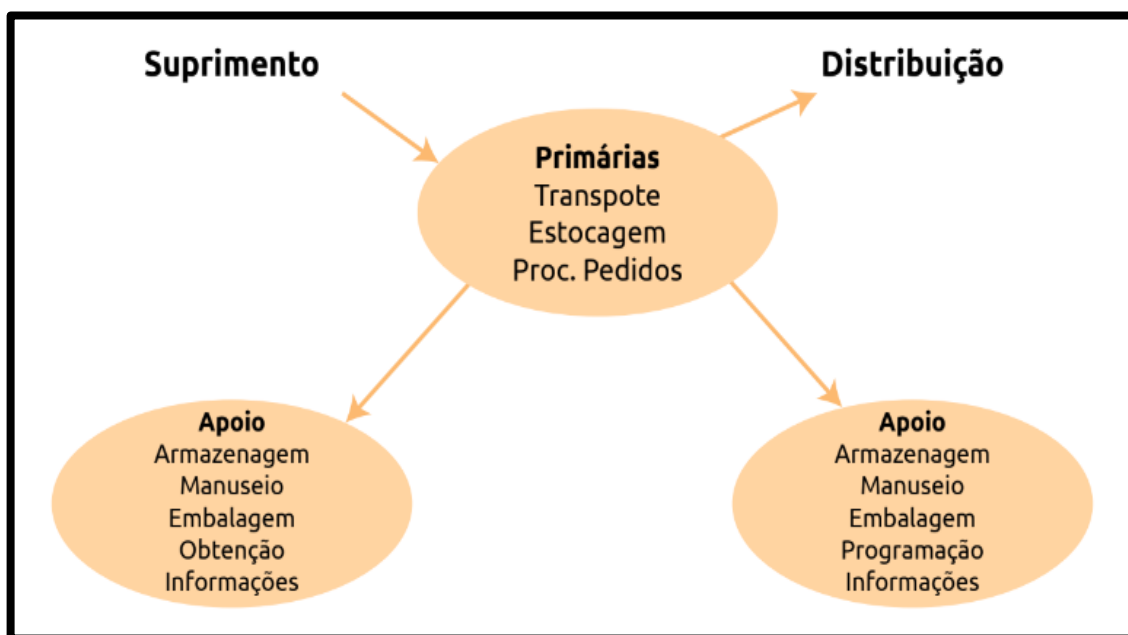
2.2. LOGÍSTICA

Conforme Benjamim Moura (2006) pode-se dizer que a logística é um processo que visa gerir os fluxos de produtos, de serviços e de quaisquer informações associadas entre o fornecedor e o cliente, sejam eles finais ou intermediários ou vice-versa. Levando aos clientes onde quer que estejam, os produtos e serviços os quais necessitam em melhores condições possíveis.

A logística é vital para os consumidores, para as organizações e para a economia em geral, por uma multiplicidade de razões, nomeadamente por haver uma grande dispersão geográfica de fornecedores e clientes, com a consequente necessidade de compatibilização da oferta (i.e., para quem? que quantidade de produto/serviço vai ser produzida ou vendida?) com a procura (i.e., que produto/serviço vai ser solicitado? Quando? quanto?), proporcionando aos clientes os bens e serviços que precisam e assegurando às empresas o escoamento da sua produção, para além, como é óbvio, do abastecimento de matérias-primas e outros inputs utilizados nas operações de produção. (MOURA, Benjamim. Logística: Conceitos e Tendências. p.15, 16).

Na Figura 1 temos um esquema que traz as atividades logísticas existentes tanto na entrada dos produtos, o suprimento físico, quanto na saída, sua distribuição física. Observa-se que as atividades são organizadas em atividades primárias e de apoio, estando presentes em toda a cadeia logística, ou seja, suprimentos e de distribuição.

Figura 1 - Categoria das atividades logísticas



Fonte: (SEST/SENAT, 2016)

Ballou, identifica que as atividades logísticas podem se tornar complexas devido a sua grande quantidade de atividades e processos, tanto para as grandes corporações quanto para as pequenas empresas, devido a sua importância. No que se diz respeito às pequenas empresas, elas não conseguem tirar proveito das compras e remessas em grandes quantidades, ao contrário das grandes corporações (BALLOU, 2006). Além disso, elas possuem uma organização centralizada, por não haver uma grande categorização de produtos, sendo ainda sobre as atividades logísticas, não haver definição e estruturas tão claras quanto nas grandes empresas.

2.3. PEQUENAS EMPRESAS

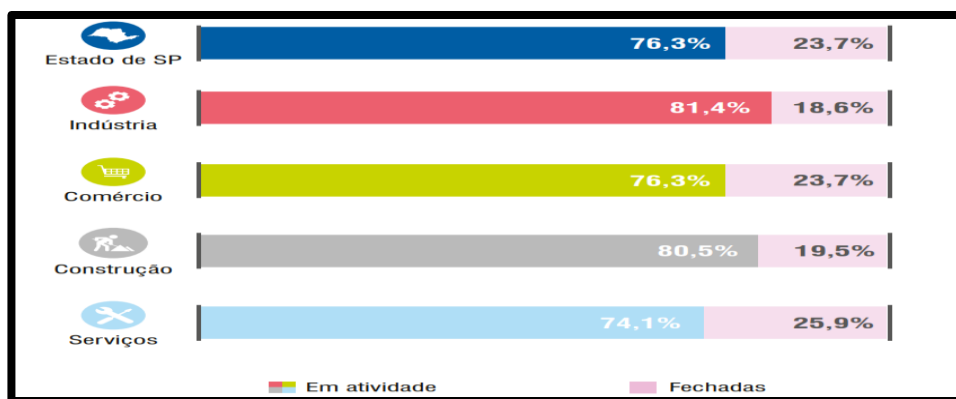
De um modo geral, as pequenas empresas são uma das bases da economia brasileira juntamente com microempresas, devido a sua capacidade de gerar empregos ou pela grande quantidade descentralizada de estabelecimentos e negócios abertos de maneira desconcentrada geograficamente (KOTESKI, 2004). Assim, as pequenas empresas têm um papel muito importante na economia mundial, pois geram inúmeros empregos, além de serem responsáveis pela maior parte da movimentação de recursos na formação do PIB dos países.

Segundo site oficial do SEBRAE (2018), uma das maiores entidades voltada para o empreendedorismo e apoio a pequenas e microempresas do Brasil, apenas em 2018 havia cerca de 6,4 milhões de estabelecimentos. Desse total, 99% eram micro e pequenas empresas. Que correspondem a 52% dos empregos com carteira assinada no setor privado 16,1 milhões.

Porém, a manutenção dessas empresas é uma tarefa difícil, no âmbito de sua administração e controle de negócios, o que leva a uma baixa expectativa de vida para esse tipo de empreendimento, tornando o investimento um tanto quanto arriscado. Ainda há outros fatores que contribuem para estas dificuldades, problemas entre sócios, crises econômicas, falta de um ambiente de negócios, alta concorrência e carga tributária (SALES; BARROS; PEREIRA, 2021).

Na Figura 2, tem-se um gráfico que mostra a taxa de sobrevivência das pequenas empresas da cidade de São Paulo, uma das cidades mais populosas do Brasil (IBGE, 2020). Onde 76,3% significa que aproximadamente 1 a cada 4 empresas registradas no CNPJ fecha em menos de 2 anos.

Figura 2 - Taxa de sobrevivência de pequenas empresas



Fonte: (SEBRAE-SP, 2016)

Entretanto, novas estratégias para a diminuição das taxas de mortalidade de pequenas empresas são criadas, como a expansão para a internet (DA SILVA, 2018), transformar o seu foco de vendas físicas e de alcance limitado em um sem fronteiras, se adaptando a uma nova segmentação: o *e-commerce*.

2.4. E-COMMERCE

Como dito por Silva (2018), *e-commerce* é um termo que começou a ser popularmente utilizado no Brasil no início do século XXI, referindo-se às vendas e negócios feitos virtualmente para seus consumidores, depois disso, foi criando ramificações de tipos de *e-commerce*. Para seu funcionamento, há a necessidade de utilizar camadas da internet, intranet entre outras redes, portanto o uso de softwares e tecnologias tornam-se indispensáveis, como o ERP, para automatização e modelagem dos processos de negócios, compartilhando informações para outras áreas, entre elas a logística (LAUDON; LAUDON, 2004).

Consequentemente, as empresas notaram a necessidade de se adaptar ao *e-commerce*, que desde seu início, sempre foi uma alternativa para agilizar transações. Atualmente abrange, não apenas as vendas de produtos, mas também serviços de vários ramos do negócio (MENDONÇA, 2016). E um dos fatores que são ponderadas no *e-commerce* é o processo logístico, que por sua vez possui elementos chave em sua formação, entre eles a gestão e monitoramento.

2.5. GESTÃO E MONITORAMENTO

A gestão e o monitoramento são processos importantes, não somente no meio da indústria e comércio, mas em todas as áreas onde ocorrem a necessidade de um processo eficiente e seguro. Para manipular de maneira eficiente a cadeia de suprimentos.

A maior organização mundial de profissionais e acadêmicos da área, o *Council of Supply Chain Management Professionals* (CSCMP, 2010) define «Logística ou Gestão Logística como a parte da Cadeia de Abastecimento que é responsável por planejar, implementar e controlar o eficiente e eficaz fluxo direto e inverso e as operações de armazenagem de bens, serviços e informação relacionada entre o ponto de origem e o ponto de consumo de forma a ir ao encontro dos requisitos/necessidades dos clientes». (DE CARVALHO, José Mexia Crespo; CARDOSO, Eduardo Gomes. Logística. Sílabo, 2002.)

Ainda nesta linha de pensamento, seguindo a definição dada pela *Council of Supply Chain Management Professionals* (CSCMP, 2010) ela define as atividades logísticas como, *inbound* e *outbound* em termos de transporte, como

entradas e saída, gestão de frota, armazenamento e distribuição. Observando a importância dessas atividades, defere-se como ela pode ser complexa a aplicar ou executar em empresas que podem não ter um domínio sobre a sua execução, necessitando de um meio robusto para fazer tal gerenciamento. Atualmente, atividades complexas passaram a ser otimizadas para utilização a partir de um aplicativo mobile. Portanto, a otimização destas atividades logísticas em aplicativos mobile tais como o gerenciamento e monitoramento são capazes de auxiliar em sua execução.

2.6. APLICATIVO MOBILE

O mercado de dispositivos mobile cresce cada vez mais e estudos apontam que apenas em 2021, 3,85 bilhões de pessoas utilizam dispositivos mobile (StrategyAnalytics, 2021), ou seja, aproximadamente metade da população mundial. Atualmente, mais usuários comuns possuem acesso à várias ferramentas possíveis do dispositivo, como mensagens instantâneas, e-mails, jogos, integrações com outros dispositivos da casa, entre outras funções, todas elas através de aplicativos.

Os aplicativos mobile, são softwares desenvolvidos para ser executado em dispositivos móveis, podendo satisfazer funções específicas para cada objetivo, um diferencial é que pode ser baixado e acessado de qualquer dispositivo. Vinculado a isto, o mercado corporativo tem buscado incorporar os aplicativos mobile ao seu cotidiano para agilizar negócios e fazer a integração dessas aplicações ao seu sistema *backend* (LECHETA, 2013). Onde as empresas buscam a mobilidade e o lucro que podem adquirir através dessas aplicações.

Desta forma, os aplicativos mobile podem ser utilizados também de forma online, havendo uma integração das informações recebidas de um servidor seguro da própria empresa e fazendo uma sincronização em tempo real. Um exemplo simples, pode ser representado na Figura 3, onde mostra um usuário utilizando um aplicativo para realizar um pedido.

Figura 3 - Usuário realiza pedido por um aplicativo



Fonte: (Ana Marques/TechTudo, 2020)

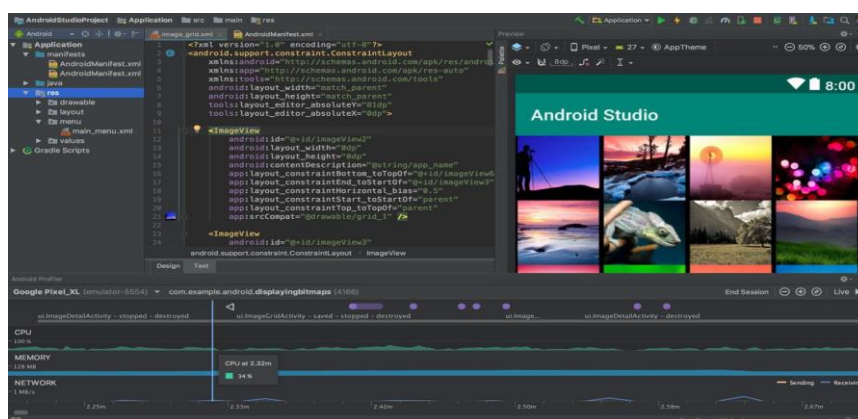
2.7. FERRAMENTAS DE DESENVOLVIMENTO DO SOFTWARE

2.7.1. *Android Studio*

Para o desenvolvimento da solução proposta, será utilizado a IDE (ambiente de desenvolvimento integrado) mais apropriado para o desenvolvimento de aplicações mobile para a plataforma Android o Android Studio (*Developer Android*, 2021), baseado em outra IDE IntelliJ IDEA. Além de editor de código, o ambiente oferece recursos para aumentar a produtividade na criação de aplicações Android. Foi escolhida devido a sua integração e compatibilidade com a plataforma Android.

Na Figura 4, observam-se as inúmeras funcionalidades presentes no IDE. Que será importante para o desenvolvimento da aplicação.

Figura 4 - IDE executando teste de performance no emulador



Fonte: (DEVELOPERS, 2021)

2.7.2. Android

A aplicação desenvolvida será projetada para executar em sistema Android uma das maiores plataformas para dispositivos móveis presentes em smartphones do mundo, cerca de 2,5 bilhões de dispositivos (CCM). Além de oferecer diversas funcionalidades, segurança e uma ótima integração com o Google, é uma ferramenta de código aberto, portanto gratuita para a criação de aplicações, o que viabiliza o desenvolvimento da solução proposta de maneira gratuita (ANDROID, 2021).

O Android funciona essencialmente como qualquer outro sistema operacional para os dispositivos móveis, fornecendo funções para gerenciar processos de aplicativos e para trabalhar com os recursos de hardware presentes no dispositivo. É interessante que o Android permite que os usuários vinculem sua conta do Google ao dispositivo e acesse seus dados nos aplicativos padrões do sistema.

Possui ainda a Google Play Store, que é uma loja virtual onde os usuários podem baixar e instalar outros aplicativos desenvolvidos por terceiros, além de permitir a publicação de aplicações recém-desenvolvidas.

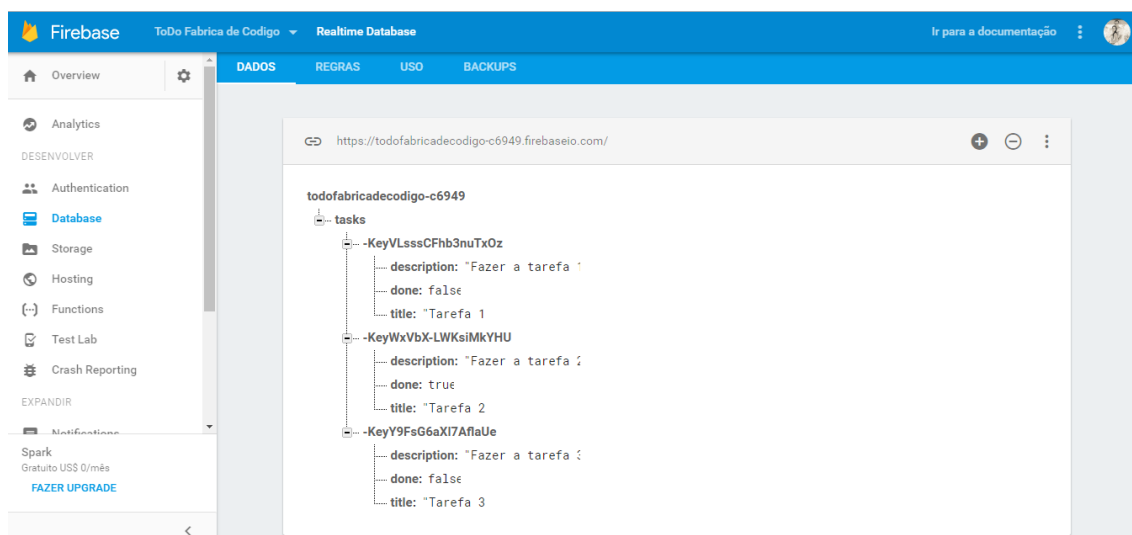
2.7.3. Firebase

Para um acompanhamento em tempo real das informações pelo aplicativo, segurança e integridade desses dados, será necessário utilizar um banco de dados online. Nesse sentido, o uso do *Firebase* contribui de maneira

significativa para um banco de dados seguro. A plataforma oferece todo o suporte necessário no *backend* da aplicação, funcionando como um *Backend as a Service (BaaS)* (FIREBASE, 2017). Além de permitir seu uso e implementação com várias linguagens, como Java, C++ e a que será utilizada no desenvolvimento deste projeto *Kotlin*.

O *Firebase* é um banco de dados *NoSQL* hospedado na nuvem. Com ele, é possível armazenar e sincronizar dados entre os seus usuários em tempo real. (FIREBASE BLOG, 2017). Além de possibilitar um console para gerenciar o banco de dados disponibilizado conforme a figura 5.

Figura 5 - Console do *Firebase* administrando um banco de dados



Fonte: (FRANCO, 2018)

O *Firebase* é multiplataforma, podendo ser voltado não apenas para plataformas *Android*, *iOS* ou móveis em geral, mas pode ser consumido através da web também, permitindo soluções para diversos meios (REMESSA ONLINE, 2021). Oferece um serviço gratuito a quem está iniciando ou possui uma baixa demanda, mas oferece planos para usos mais avançados se necessário.

2.7.4. NoSQL

Os Bancos de Dados *NoSQL* são projetados para tratar modelos de dados específicos com esquemas flexíveis para a criação de aplicativos modernos. São amplamente reconhecidos por sua facilidade de desenvolvimento, funcionalidade e desempenho em escala (AWS, 2022). Utilizam-se de várias maneiras de acessar e manipular dados, sempre de maneira flexível, que só são

possíveis devido o relaxamento de restrições de consistência, que existem em outros bancos.

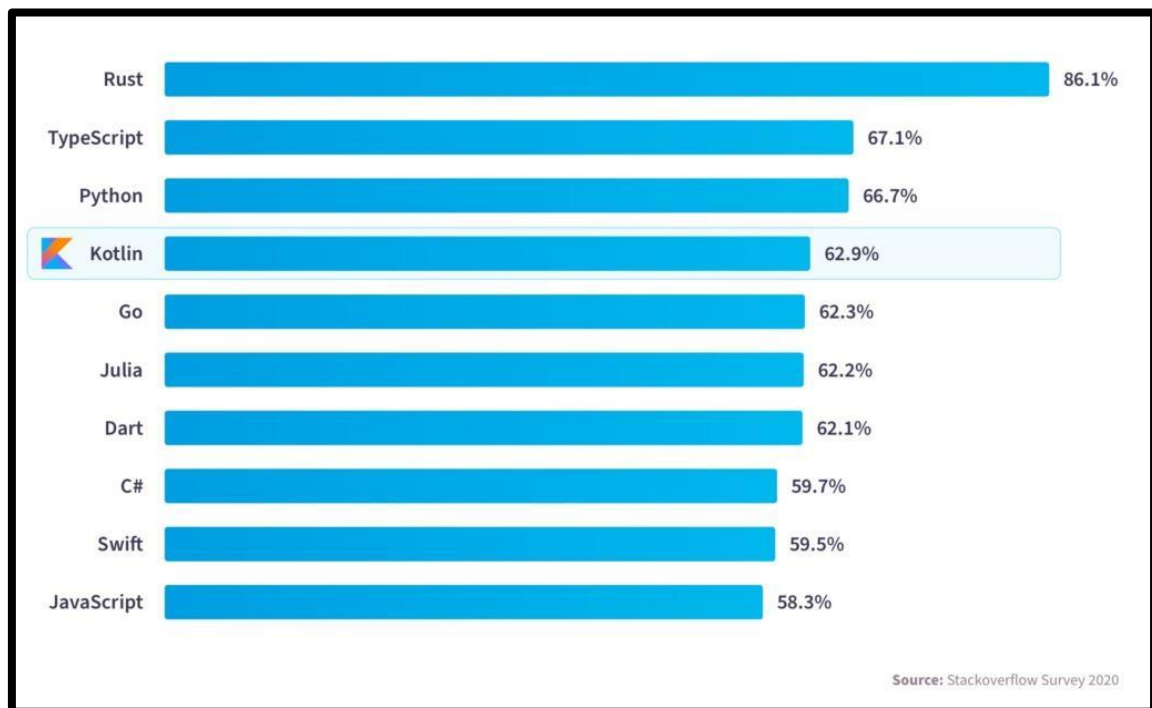
2.7.5. Kotlin

Será utilizada a linguagem *Kotlin* para desenvolvimento da aplicação, devido à sua tipagem moderna, concisão, segurança, interoperável com Java e outras linguagens, fornece várias maneiras de reutilizar código em várias plataformas para uma programação eficiente. Além de ser uma das linguagens criadas para desenvolvimento de aplicações para a plataforma Android (KOTLINLANG.ORG, 2022).

Kotlin é uma linguagem projetada para orientação a objeto de força industrial, uma melhoria da linguagem Java, possui a capacidade de ser Interoperável com Java, facilitando qualquer necessidade de reutilização de código. Em 2017 a Google anunciou suporte à linguagem *Kotlin*, impulsionando ainda mais sua popularidade entre desenvolvedores Android.

Na Figura 6, mostra a porcentagem de pessoas que adotaram a linguagem *Kotlin* como preferência para desenvolvimento, ficando entre as 5 linguagens mais adotadas pelos desenvolvedores,

Figura 6 - Popularidade da linguagem *Kotlin*



Fonte: (StackOverFlow, 2020)

2.7.6. SCRUM

A técnica para cumprimento de cronograma a ser utilizada será o SCRUM, onde há divisão de tarefas complexas em outras tarefas menores com datas limites e revisões das chamadas sprints. Por ser uma metodologia ágil, que visa principalmente a reduzir o tempo de entrega do produto a ser desenvolvido e se adaptar a qualquer mudança ou contratempo que venha a existir durante as etapas de produção.

O Scrum, é um framework onde é possível se adaptar a problemas complexos no meio de produção, conseguindo assim entregar de maneira produtiva e criativa produtos de maior valor possível (SCHWABER; SUTHERLAND, 2017). De modo geral, o Scrum visa a conclusão ou entrega de um produto de maneira ágil, podendo se adaptar ao decorrer do processo de produção.

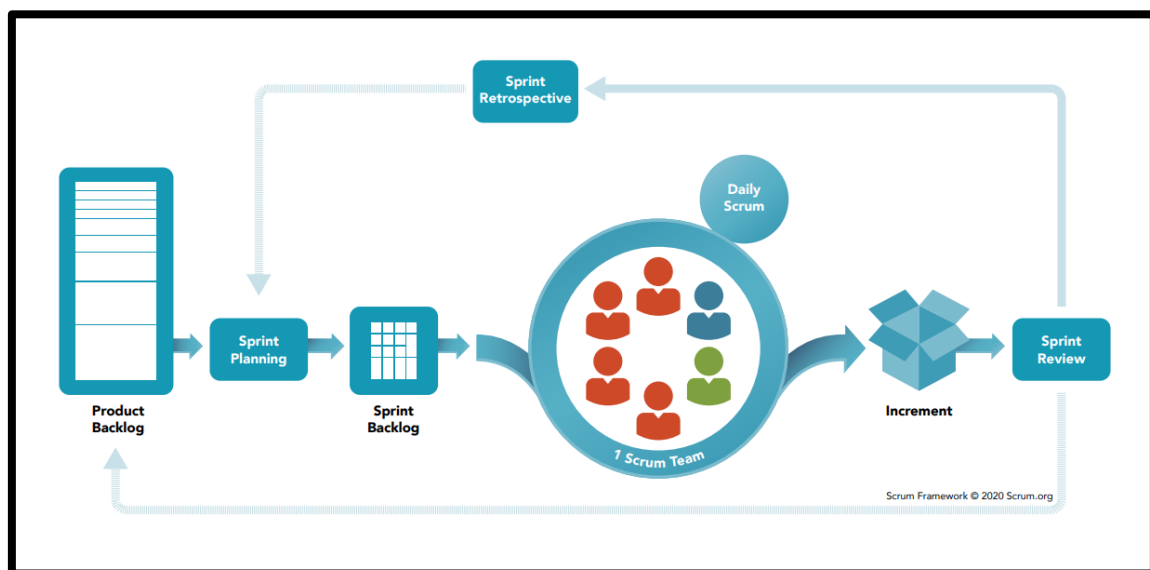
Para utilizar o framework Scrum, é necessário ter um Scrum Master, que irá promover um ambiente que contém o Product Owner e o Scrum Team onde:

- *Product Owner* – Ordena a divisão de um trabalho ou requisito complexo em um *Product Backlog*.

- Scrum Team – Transforma este trabalho em um incremento de valor em uma sprint.
- Ocorre a avaliação de resultados juntamente com os Stakeholders, além de uma autoavaliação onde se planejam para a próxima sprint.

Na Figura 7, observa-se o ciclo de vida do SCRUM, partindo desde as definições do *Product Backlog*, *Sprint Backlog* e Incremento até chegar ao Sprint Review onde se reinicia o ciclo.

Figura 7 - Ciclo de vida do framework SCRUM



Fonte: (SCRUM.ORG, 2020)

2.8. TRABALHOS RELACIONADOS

Existem sistemas que já propõem soluções para gerenciar entregas e pequenas empresas, nesta seção será apresentado alguns trabalhos e sistemas que irão servir como base para desenvolvimento da aplicação proposta no trabalho.

2.8.1. Sistema de Pizzaria FW PIZZA

Nesta monografia WEIBER (2017), propõem o desenvolvimento de um sistema de informação para pizzarias, sendo composto por módulos para controle de transações, cadastros de clientes, produtos, gerenciamento de entregas e mesas, contando ainda com um módulo para auxiliar na tomada de

decisão, através da apresentação de gráficos e relatórios de maneira a auxiliar o gerente.

Na Figura 8, temos o Dashboard do sistema, sendo possível observar algumas funcionalidades citadas, como cadastros, relatórios, pagamentos, pedidos de entregas, entre outros.

Figura 8 - Dashboard do Sistema FW Pizza



Fonte: (WEIBER, 2017)

Optou-se por consultar este trabalho, pois implementa algumas funcionalidades que estarão presentes no desenvolvimento da aplicação proposta pela presente pesquisa, entre elas a processamento de pedidos de entregas, cadastros de clientes e relatórios.

2.8.2. Aplicação Web de Gerenciamento de Entregas de Transportadoras - SHIPPING

Nesta monografia SCHARF (2018), apresenta um sistema web de gerenciamento de entregas para transportadoras, com o objetivo de auxiliar as transportadoras a fazer a distribuição de produtos, visando suprir o abastecimento tanto para consumidores, quanto para empresas. Pensando ainda, na segurança, monitoramento e otimização dessa distribuição.

Na Figura 9, observa-se a aplicação web em execução, utilizando a função de rastreo do entregador, utilizando um sistema de cadastro de históricos.

Figura 9 - Sistema SHIPPING utilizando a função de rastreio do entregador

The screenshot displays the SHIPPING system interface. On the left is a dark sidebar with a 'Shipping' logo and a list of menu items: 'Cadastrados', 'Pessoas', 'Espaços', 'Movimentações', 'Patrimônio', and 'Relatórios'. The main area has a light gray background with the title 'Cadastro de Históricos'. Below the title is a form with the following fields: 'Código:' (empty), 'Descrição:' (containing 'Intervenções na pista'), 'Data:' (containing '13/12/2018'), 'Latitude:' (containing '-26.2624687'), 'Longitude:' (containing '-52.67661779999995'), and 'Viagem:' (a dropdown menu with '1' selected). At the bottom of the form are 'Limpar' and 'Salvar' buttons. Below the form is a button labeled 'Obter Localização'. A small 'Sair' button is located in the top right corner of the header.

Fonte: (SCHARF, 2018)

Através da consulta a este material, será possível, ter uma base sólida com o que se diz respeito a coleta de requisitos, a maneira que será implementada seguirá um modelo similar, além de o mesmo utilizar recursos tecnológicos semelhantes para a coleta da localização do usuário.

2.8.3. TudoEntregue

Esta ferramenta desenvolvida pela Active Corp, que visa facilitar a o controle das entregas e coletas integrado com uma torre de controle que utiliza ainda uma api para integrar a versão web com mobile, dispõe de um período de testes gratuito, logo após é necessário assinar um dos planos de uso.

Baseando-se em algumas funcionalidades desta ferramenta, foi analisado como poderia implementar atividades tão complexas de uma maneira mais fácil. Visando a usabilidade que o usuário irá ter com a aplicação implementada.

2.8.4. Comparativo

A tabela 1 abaixo, mostra algumas características dos sistemas relacionados, dois deles, são apenas para a versão web quanto a última é integrada entre as versões mobile e desktop.

Tabela 1 - Características dos sistemas apresentados

Características	FW PIZZA	SHIPPING	TudoEntregue	Meu Projeto
1. Mobile			x	x
2. Desktop	x	x	x	
3. Gratuito	x	x		x
4. Armazenamento em Banco de Dados		x	x	x

3 PROJETO DO SISTEMA

3.1. INTRODUÇÃO

O presente trabalho visa projetar um sistema que ajuda na agilização do gerenciamento de atividades logísticas, mais precisamente nas atividades primárias como: monitoramento e planejamento de rotas, processamento de pedidos e controle de entregas, como foco em pequenas empresas, podendo utilizar funcionários fixos ou temporários.

A aplicação poderá ser acessada através da criação de um administrador, que por sua vez terá acesso a inclusão e cadastro de sua empresa, entregadores, encomendas e clientes. Poderá ser acessado pelo entregador que realizará a entrega atribuída a ele, além de atualizar os status de entregas. Contando ainda com um módulo para o cliente, monitorar o status de sua encomenda, através de um código gerado para o rastreio.

3.2. MODELAGEM DO SISTEMA

A tabela 2 a baixo apresenta os requisitos funcionais (RF) que guiaram na implementação do sistema.

Tabela 2 - Requisitos Funcionais

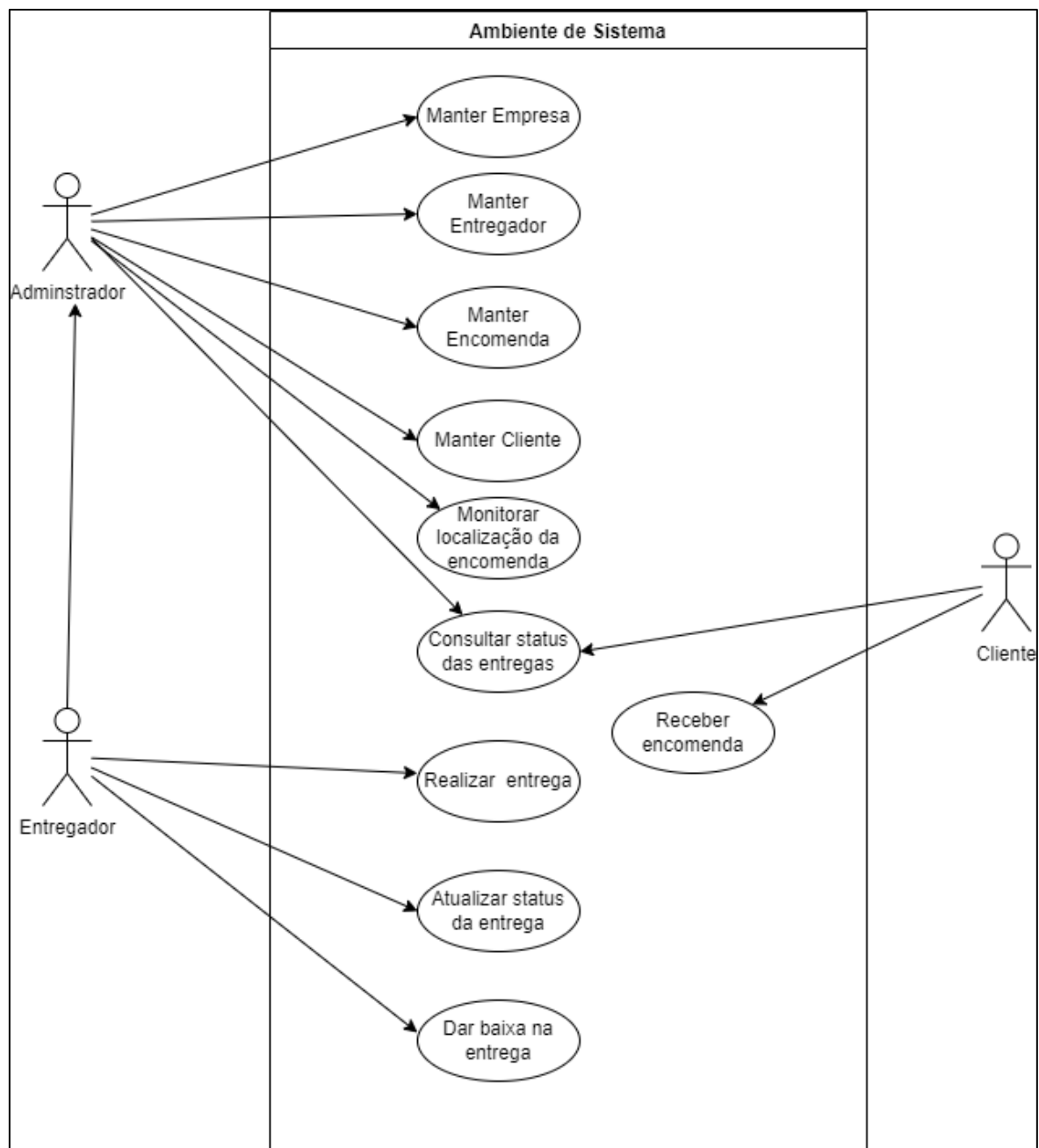
Id	Nome	Responsável	Descrição
RF01	Manter Empresa	Administrador	O cadastro de Empresa será composto dos atributos (nome, telefone, e-mail, CPF/CNPJ, endereço e senha).
RF02	Manter Entregador	Administrador	O cadastro de Entregador será composto dos atributos (nome, telefone, CNH, categoria de CNH, e-mail, CPF, endereço e senha).
RF03	Manter Encomenda	Administrador	O cadastro de Encomendas será composto dos atributos (descrição, cliente origem, cliente destino, data da coleta, valor do frete e peso).
RF04	Manter Cliente	Administrador	O cadastro de Clientes será composto dos atributos (nome, telefone, e-mail, CPF/CNPJ, endereço).

RF05	Monitorar Localização da encomenda	Administrador	A consulta pela localização da entrega poderá ser feita pelo Administrador ou Cliente com código de rastreio.
RF06	Consultar Status das Entregas	Administrador e Cliente	A consulta pelo status da entrega poderá ser feita pelo Administrador ou Cliente com código de rastreio.
RF07	Receber Encomenda	Cliente	O Cliente poderá receber a encomenda e enviar uma confirmação de que recebeu.
RF08	Realizar Entrega	Entregador	O Entregador poderá realizar a entrega a partir do agendamento realizado.
RF09	Atualizar Status da Entrega	Entregador	O Entregador será capaz de atualizar os status da entrega, assim como registrar ocorrências em tempo real.
RF10	Dar Baixa na Entrega	Entregador	O Entregador poderá finalizar Entrega após a realização da mesma.

3.3. DIAGRAMA DE CASOS DE USO

A Figura 10 apresenta o diagrama de casos de uso com as funcionalidades principais do sistema, que serão executadas pelos atores Administrador, Entregador e Cliente.

Figura 10 - Diagrama de casos de uso



Fonte: O autor

3.4. DESCRIÇÃO DO CASO DE USO

A tabela 2 mostra a descrição do caso de uso manter Empresa, que terá o envolvimento do ator Administrador.

Tabela 3 - Descrição do caso de uso Manter Empresa

Caso de uso	UC01 – Manter Empresa
Descrição	O administrador poderá ter acesso às funcionalidades básicas para gerenciar a entidade empresa.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none">• O administrador deve estar cadastrado e logado.• O administrador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de empresa.4. Escolher operações a realizar.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso não esteja cadastrado, o administrador deve se cadastrar.

A tabela 4 mostra a descrição do caso de uso manter entregador, que terá o envolvimento do ator Administrador.

Tabela 4 - Descrição do caso de uso Manter Entregador

Caso de uso	UC02 – Manter Entregador
Descrição	O administrador poderá ter acesso às funcionalidades básicas para gerenciar a entidade entregador.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none">• O administrador deve estar cadastrado e logado.• O administrador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de entregador.4. Escolher operações a realizar.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso não esteja cadastrado, o administrador deve se cadastrar.

A tabela 5 mostra a descrição do caso de uso manter Encomenda, que terá o envolvimento do ator Administrador.

Tabela 5 - Descrição do caso de uso Manter Entregador

Caso de uso	UC03 – Manter Encomenda
Descrição	O administrador poderá ter acesso às funcionalidades básicas para gerenciar a entidade encomenda.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none">• O administrador deve estar cadastrado e logado.• O administrador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de encomendas.4. Escolher operações a realizar.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso não esteja cadastrado, o administrador deve se cadastrar.

A tabela 6 mostra a descrição do caso de uso manter Cliente, que terá o envolvimento do ator Administrador.

Tabela 6 - Descrição do caso de uso Manter Cliente

Caso de uso	UC04 – Manter Cliente
Descrição	O administrador poderá ter acesso às funcionalidades básicas para gerenciar a entidade Cliente.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none">• O administrador deve estar cadastrado e logado.• O administrador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de Clientes.4. Escolher operações a realizar.
Fluxo alternativo	<ol style="list-style-type: none">2. Caso não esteja cadastrado, o administrador deve se cadastrar.

A tabela 7 mostra a descrição do caso de uso Monitorar Localização da Encomenda, que terá o envolvimento do ator Administrador.

Tabela 7 - Descrição de caso de uso Monitorar Localização da Encomenda

Caso de uso	UC05 – Monitorar localização da encomenda
Descrição	O administrador deve poder monitorar a localização da entrega.
Ator	Administrador
Pré-condição	<ul style="list-style-type: none">• O administrador deve estar cadastrado e logado.• O administrador deve já ter cadastrado a empresa, entregador e o cliente.• O administrador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de Entrega.4. Selecionar entrega que deseja monitorar.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso não esteja cadastrado, o administrador deve realizar o cadastro da encomenda ou agendamento da Entrega.

A tabela 8 mostra a descrição do caso de uso consultar status das entregas, que terá o envolvimento do ator Administrador e Cliente.

Tabela 8 - Descrição do caso de uso consultar status das entregas

Caso de uso	UC06 – Consultar status das entregas
Descrição	O administrador e cliente poderá ter acesso à consulta do status da encomenda cliente.
Ator	Administrador, Cliente.
Pré-condição	<ul style="list-style-type: none">• O usuário deve estar cadastrado.• O Cliente deve ter o código de rastreio da encomenda.• O usuário deve possuir acesso à internet.
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Usuário deve inserir dados para login (e-mail, senha).2. Fazer login.3. Acessar o módulo de encomendas.4. Selecionar encomenda.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso o usuário seja o cliente, no passo 1, deve utilizar o CPF e o código de rastreio fornecido.

A tabela 8 mostra a descrição do caso de uso Realizar Entrega, que terá o envolvimento do ator Administrador e Cliente.

Tabela 9 - Descrição do caso de uso Realizar Entrega

Caso de uso	UC07 – Realizar entrega
Descrição	O entregador poderá realizar a entrega da encomenda ao cliente.
Ator	Entregador, Cliente.
Pré-condição	<ul style="list-style-type: none">• O entregador deve estar cadastrado e logado.• O entregador deve consultar os dados da encomenda.• O entregador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (CPF, senha).2. Fazer login.3. Acessar o módulo de entregas pendentes.4. Selecionar a entrega atribuída a ele.5. Realizar entrega.6. Atualizar status da entrega.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso não esteja cadastrado, o entregador deve ser cadastrado pelo administrador.

A tabela 9 mostra a descrição do caso de uso Atualizar Status da Entrega, que terá o envolvimento do ator Entregador.

Tabela 10 - Descrição do caso de uso Atualizar Status da Entrega

Caso de uso	UC08 – Atualizar status da entrega
Descrição	O entregador poderá atualizar os status da entrega
Ator	Entregador
Pré-condição	<ul style="list-style-type: none">• O entregador deve estar cadastrado e logado.• O entregador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (CPF, senha).2. Fazer login.3. Acessar o módulo de entregas.4. Atualizar o status da entrega.
Fluxo alternativo	-

A tabela 11 mostra a descrição do caso de uso Dar Baixa na Entrega, que terá o envolvimento do ator Entregador.

Tabela 11 - Descrição do caso de uso de Dar Baixa na Entrega

Caso de uso	UC09 – Dar baixa na entrega
Descrição	O entregador deverá dar a baixa da entrega após realização da mesma com sucesso.
Ator	Entregador.
Pré-condição	<ul style="list-style-type: none">• O entregador deve estar cadastrado e logado.• O cliente deve confirmar o recebimento.• O entregador deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (CPF, senha).2. Fazer login.3. Acessar o módulo de entregas.4. Confirmar a entrega.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso a tentativa de entrega falhe, o entregador deve registrar uma ocorrência da tentativa e realizar nova entrega.

A tabela 12 mostra a descrição do caso de uso Receber Encomenda, que terá o envolvimento do ator Cliente.

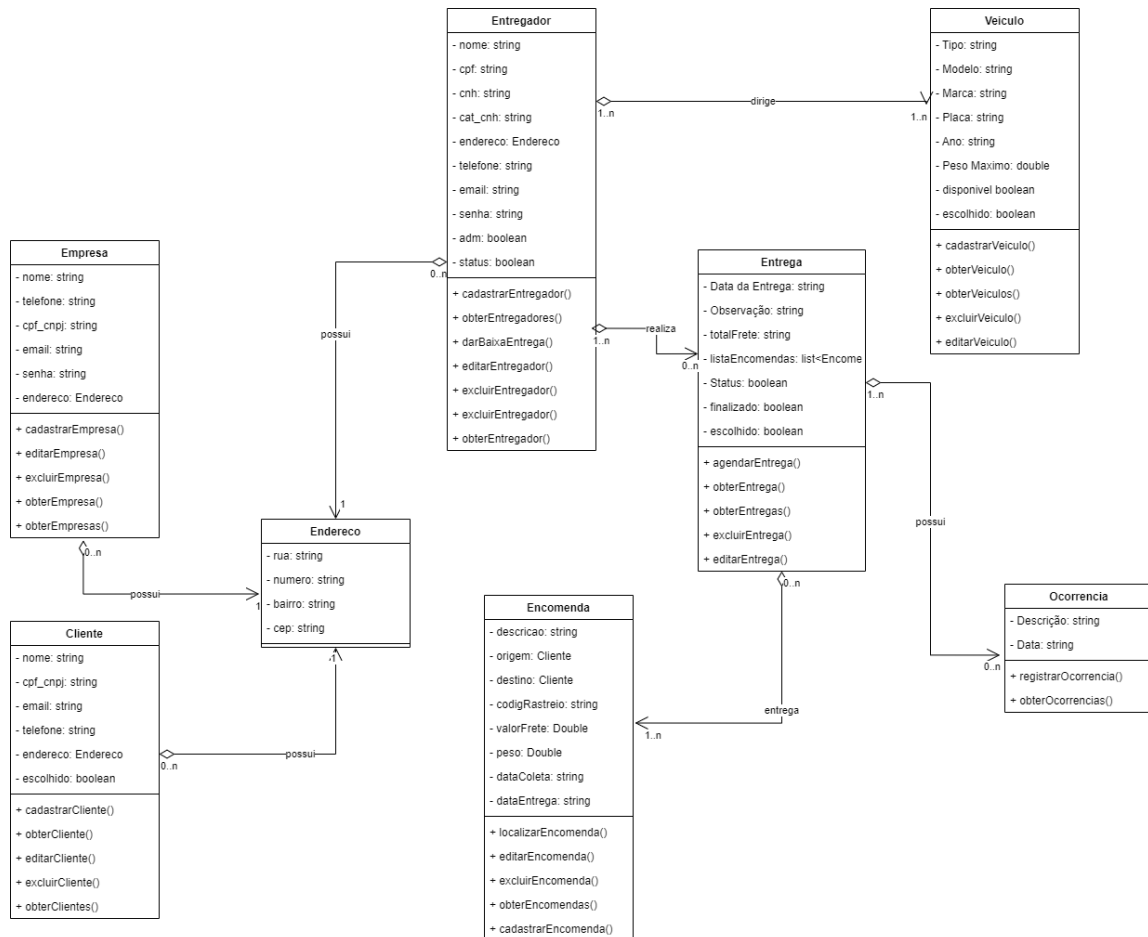
Tabela 12 - Descrição do caso de uso Receber Encomenda

Caso de uso	UC10 – Receber encomenda
Descrição	O cliente poderá confirmar o recebimento da entrega.
Ator	Cliente.
Pré-condição	<ul style="list-style-type: none">• O cliente deve estar cadastrado.• O cliente deve possuir acesso à internet
Pós-condições	-
Fluxo principal	<ol style="list-style-type: none">1. Inserir dados para login (CPF, código de rastreio).2. Acessar detalhes da encomenda e confirmar recebimento.3. Confirmar recebimento.
Fluxo alternativo	<ol style="list-style-type: none">1. Caso haja problemas no recebimento, poderá comunicar o feedback no formulário.

3.5. DIAGRAMA DE CLASSES

A Figura 11 apresenta o diagrama de classes do sistema, trazendo ainda suas relações e métodos de suas respectivas classes.

Figura 11 - Diagrama de Classes



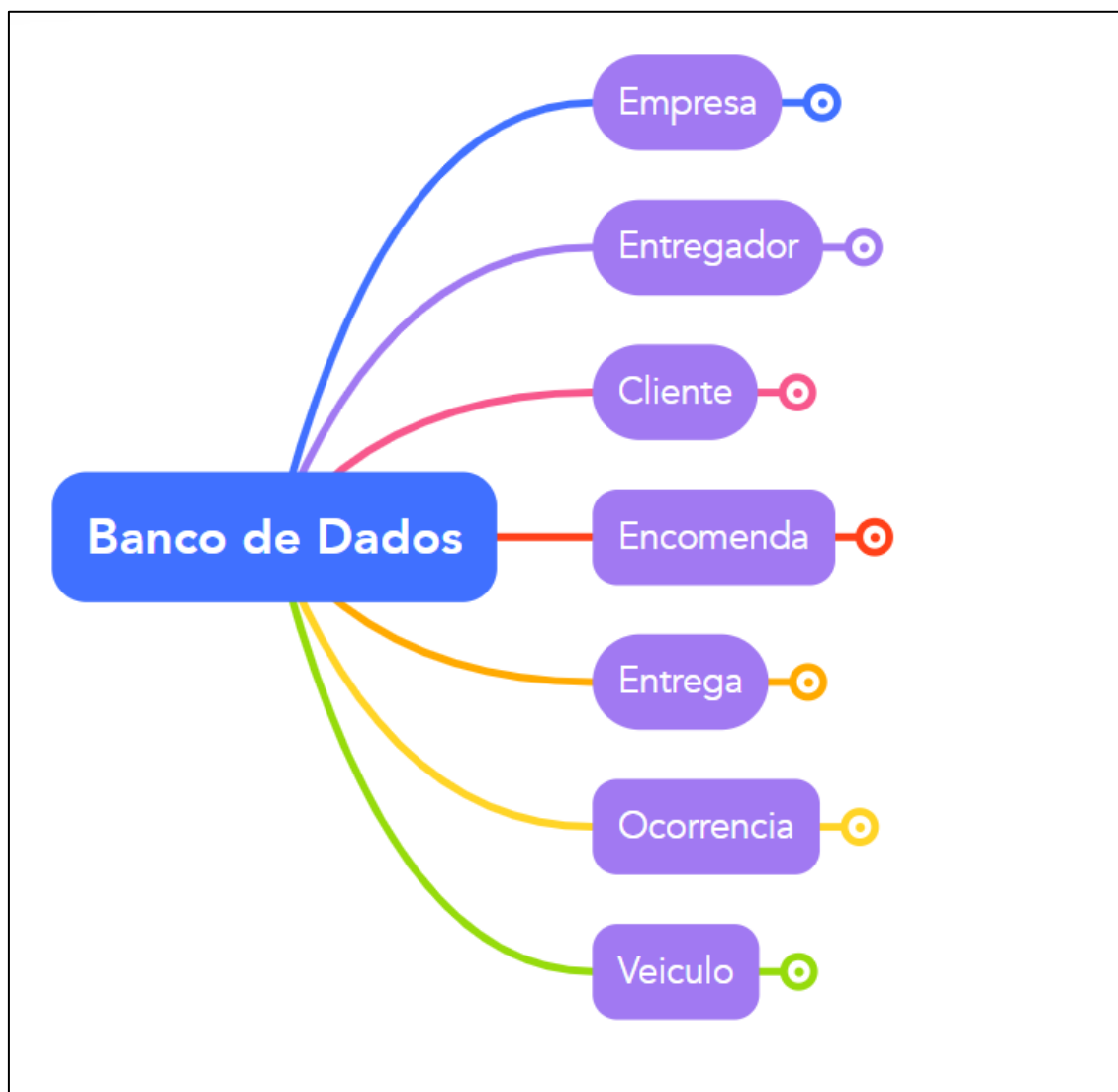
Fonte: O autor

3.6. MODELO DO BANCO DE DADOS

O modelo utilizado foi o *NoSQL*, por nesse modelo as relações entre os documentos são realizadas por referências, não sendo uma dependência direta, o que torna mais rápida e fácil a consulta desses dados.

A figura 12 mostra como as coleções ficam distribuídas dentro do banco de dados.

Figura 12 - Distribuição dos documentos dentro do Banco de Dados



Fonte: O autor

4 IMPLEMENTAÇÃO

4.1. INTRODUÇÃO

Neste trabalho foi implementado um sistema mobile para gerenciamento e monitoramento de entregas, sendo a aplicação dividida em três partes, o modulo administrativo, cliente, entregador. Onde no modulo administrativo é possível gerenciar o agendamento de entregas, cadastros de clientes, encomendas, veículos e entregadores. No modulo de cliente, é possível acessar detalhes e a localização do entregador em tempo de entrega. Quanto ao modulo do entregador é possível verificar as entregas pendentes e realizar a entrega pela rota mais conveniente apresentada pela api do Google Maps.

O aplicativo possui banco de dados online, ou seja, os dados cadastrados podem ser acessados de outro dispositivo do usuário, atualizações em tempo real entre outras, necessitando apenas de uma rede internet.

4.2. FUNCIONALIDADES DO SISTEMA

O aplicativo desenvolvido possui as seguintes funcionalidades implementadas:

- Cadastro de Empresas.
- Cadastro de Entregadores.
- Cadastro de Clientes.
- Cadastro de Veículos.
- Cadastro de Encomendas.
- Agendamento de Entregas.
- Consulta de dados cadastrados (Empresas Entregadores, Clientes, Veículos, Encomendas e Entregas)
- Rastreio de Localização de Encomenda.
- Consulta de Status da Entrega/Encomenda.
- Funcionalidades diferentes para os três módulos disponíveis.

4.3. APRESENTAÇÃO DO SISTEMA

Neste capítulo será apresentado a aplicação desenvolvida tal as funcionalidades disponíveis.

O aplicativo funciona como um gerenciador e monitor de entregas, possibilitando um controle de informações em tempo real, auxiliando no agendamento de entregas até a confirmação do recebimento pelo cliente, além de permitir que o Administrador possa consultar a localização atual do entregador ou encomenda.

4.3.1. Configuração, Controle de Acesso e Modulo da Transportadora

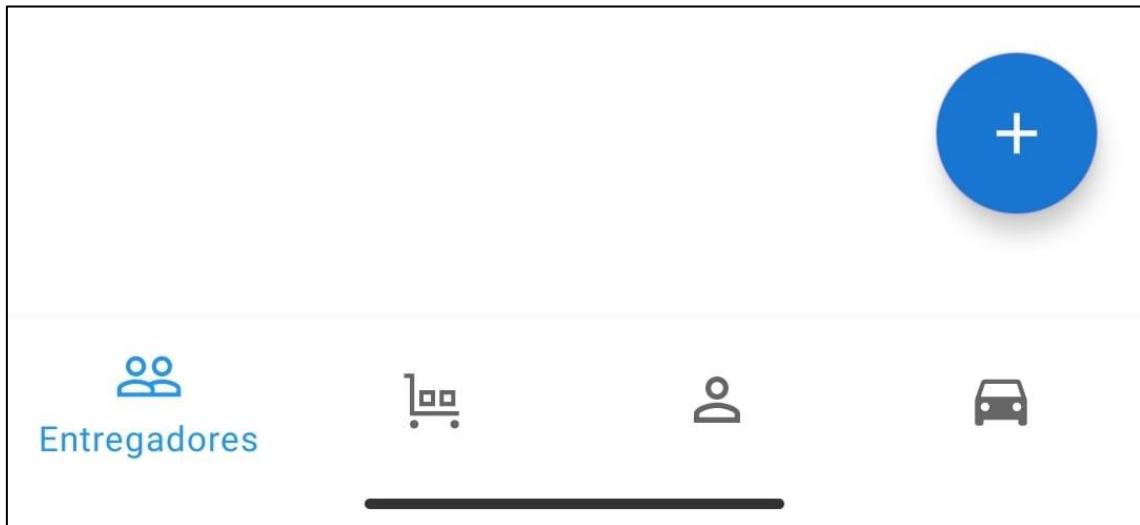
Ao acessar o aplicativo, será apresentado três módulos de acesso conforme a figura 13, deve-se criar primeiramente a empresa selecionando o modulo de Transportadora, para isso é preciso realizar o cadastro da empresa, criando um usuário com e-mail e senha, para acessar o modulo principal. Após realizar o cadastro, deve-se realizar o login com o e-mail e senhas criados, validados o e-mail e senha, o usuário tem acesso a tela principal do modulo que possui um menu de navegação na parte inferior da tela conforme a figura 14.

Figura 13 - Módulos de Acesso do aplicativo



Fonte: O autor

Figura 14 - Menu de navegação



Fonte: O autor

Através de cada ícone do menu é possível chegar em uma sessão específica, como Entregadores, Entregas, Clientes e Veículos onde os mesmos terão um botão flutuante de onde poderá acessar os formulários de cadastro de cada uma das respectivas sessões conforme as figuras 15 e 16.

Figura 15 - Formulários de cadastros

← Cadastrar Entregador	Cadastrar Cliente
<input type="text" value="Nome"/>	<input type="text" value="Nome"/>
<input type="text" value="Telefone"/>	<input type="text" value="CPF"/>
<input type="text" value="Email"/>	<input type="text" value="Telefone"/>
<input type="text" value="CPF"/>	<input type="text" value="Email"/>
<input type="text" value="CNH"/>	<input type="text" value="Endereço"/>
<input type="text" value="Categoria da CNH"/>	<input type="text" value="Bairro"/>
<input type="text" value="Endereço"/>	<input type="text" value="Número"/>
<input type="text" value="Bairro"/>	<input type="text" value="CEP"/>
	<input type="button" value="SALVAR"/>
	<input type="button" value="CANCELAR"/>

Fonte: O autor

Figura 16 - Formulários de cadastro e agendamento

Cadastrar Veículo

Categoria

Modelo

Marca

Placa

Ano

Peso Max.

SALVAR CANCELAR

Agendar Entrega

Previsão de Entrega

Descrição

Veículos

Placa: JAKAKKAK
Peso Max.: 120.0 Kg
Modelo: Pop 100

Placa: ASD232KK
Peso Max.: 350.0 Kg
Modelo: Palio

Placa: WEV12324
Peso Max.: 2000.0 Kg
Modelo: Yveco

Encomendas

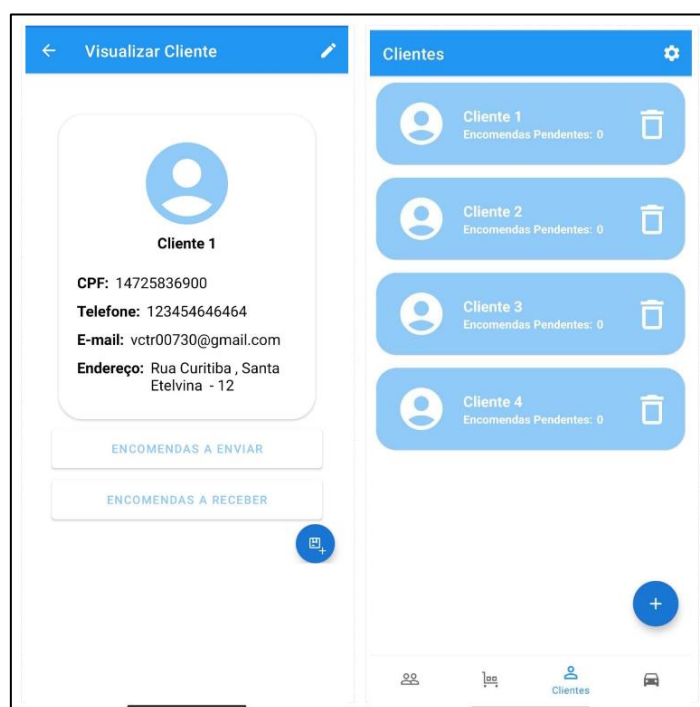
Peso Total: Kg
Total: R\$
Qtd. Encomendas:

SALVAR CANCELAR

Fonte: O autor

Depois de cadastrados é possível visualizar as informações cadastradas ou se necessário realizar a edição. Indo para a sessão de clientes, é possível ver uma lista de clientes cadastrados, e ao clicar sobre o item pode-se ver informações do cadastro além de encomendas cadastradas que devem ser enviadas ou recebidas conforme a figura 17.

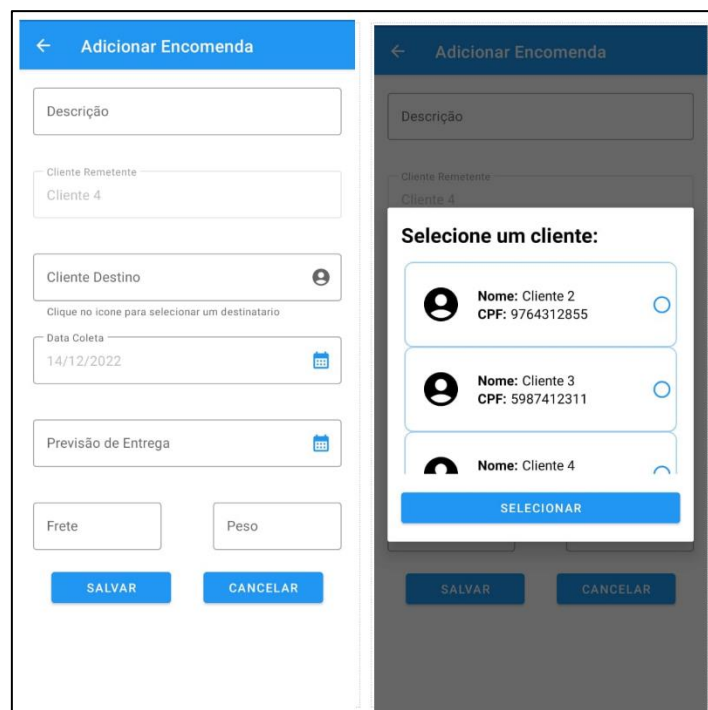
Figura 17 - Visualização e lista de clientes



Fonte: O autor

A partir da tela de visualização é possível cadastrar uma nova encomenda a ser enviada pelo cliente conforme a figura 18.

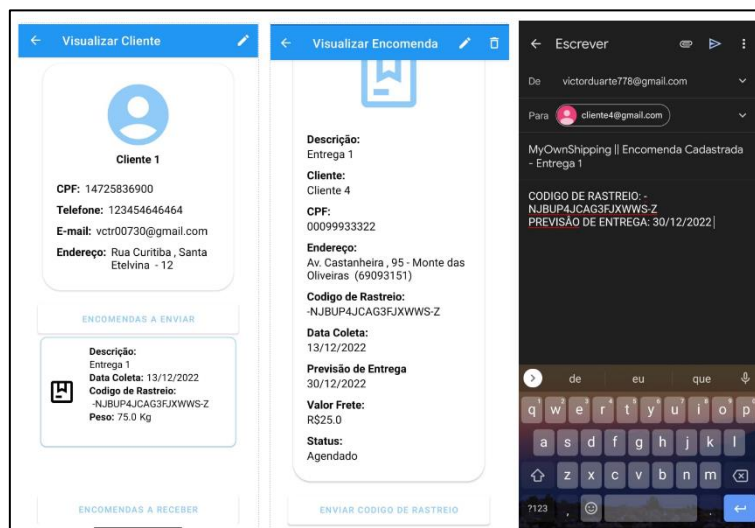
Figura 18 - Cadastro de encomenda e Seleção de cliente



Fonte: O autor

Ao cadastrar uma encomenda, é possível visualizar informações sobre ela, além de enviar um e-mail de notificação com o código de rastreio da encomenda para consultas do cliente conforme a figura 19.

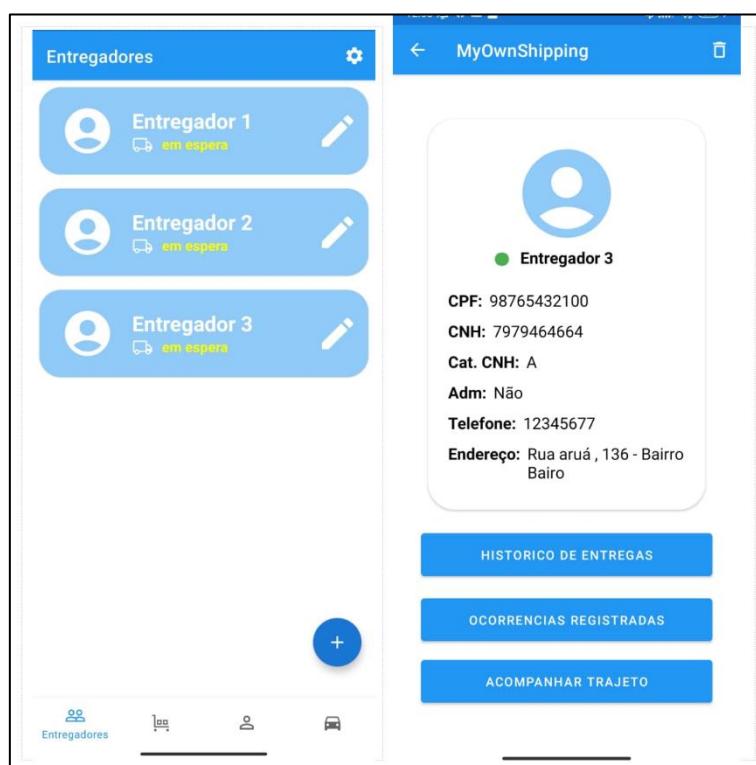
Figura 19 - Visualização e envio do código de rastreio para o cliente



Fonte: O autor

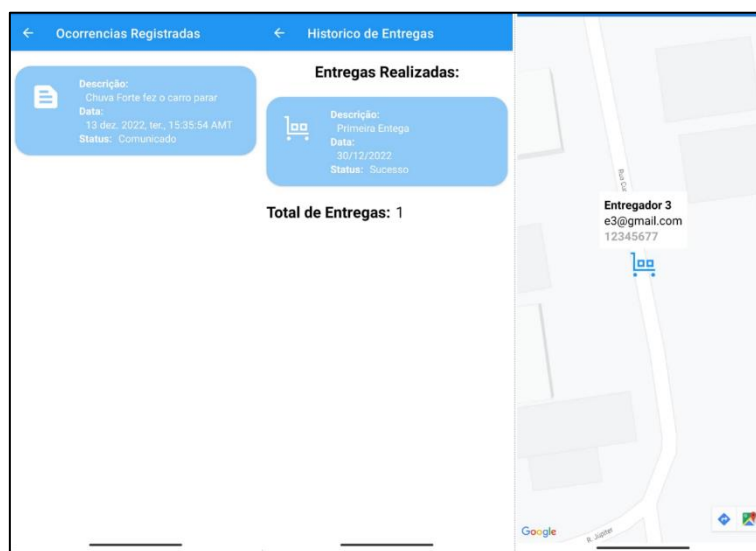
Voltando a tela de entregadores, é possível ver os entregadores e seus status “em espera” ou “em rota”, ao clicar em um entregador é possível visualizar informações sobre o mesmo e consultar entregas realizadas e ocorrências registradas, caso o entregador esteja com o status de “em rota” é possível visualizar a localização do entregador conforme as Figuras 20 e 21.

Figura 20 - Sessão e visualização de entregador



Fonte: O autor

Figura 21 - Opções da visualização do entregador

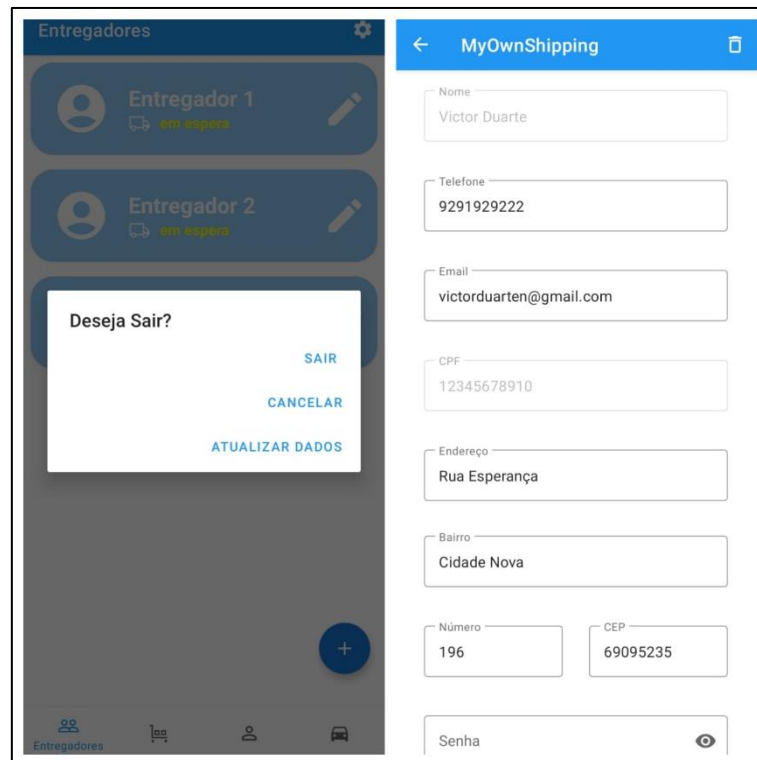


Fonte: O autor

Caso haja necessidade no menu principal pode-se acessar as operações de gerenciamento da empresa, no caso o usuário que faz login como empresa tem a permissão para fazer logout, editar ou excluir empresa clicando no ícone de engrenagem conforme a figura 22, o usuário que entra como administrador

pode realizar as mesmas operações disponíveis no módulo, porém sem acesso as operações da empresa.

Figura 22 - Alerta com ações disponíveis e edição de dados da Empresa

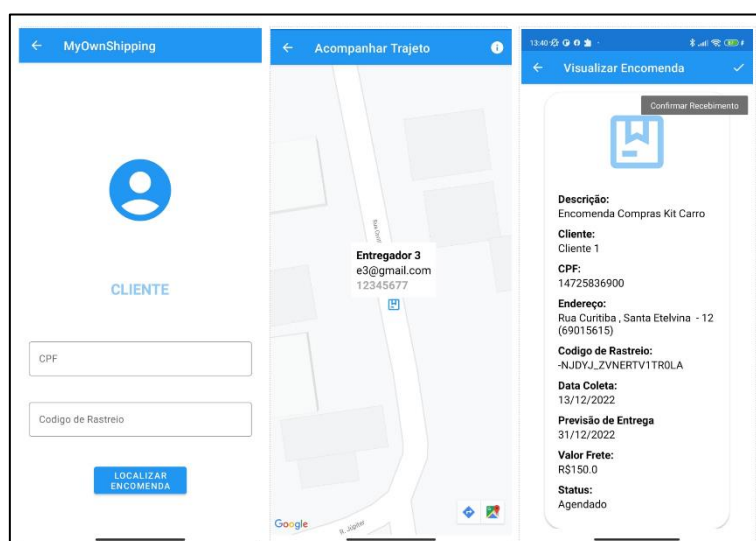


Fonte: O autor

4.3.2. Módulo do Cliente

No módulo de cliente, é possível localizar a encomenda com o CPF e o código de rastreamento recebido. Caso o entregador esteja em rota, será permitido ver a localização atual, se não o cliente será redirecionado para a tela de informação da encomenda, onde ele poderá confirmar recebimento ou visualizar informações da sua encomenda, conforme a figura 23.

Figura 23 - Modulo do Cliente

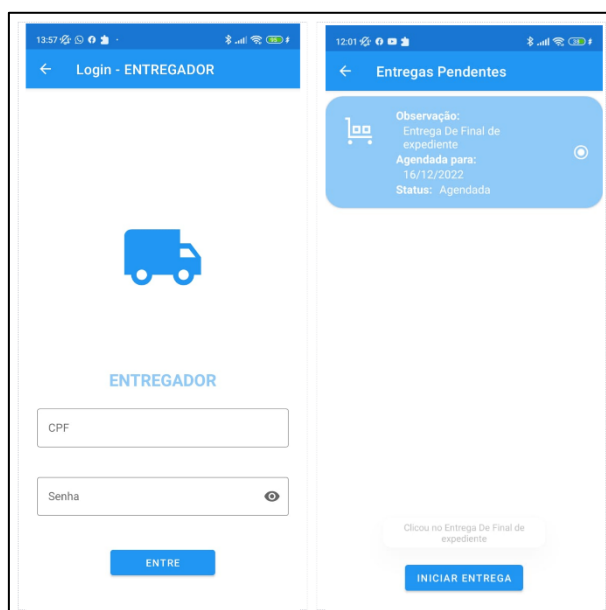


Fonte: O autor

4.3.3. Modulo do Entregador

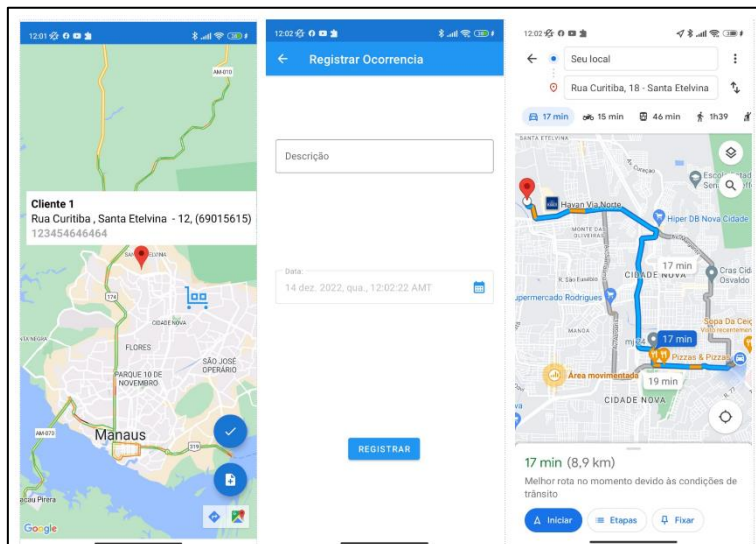
O último módulo é o de Entregador, onde deve ser realizado o login com o CPF e senhas cadastrados por um administrador, logo após o entregador tem acesso a tela de entregas pendentes, ao selecionar uma para realizar a entrega ele é redirecionado para a tela de trajeto para iniciar a rota provida pelo Google Maps, além de poder registrar uma ocorrência e concluir entrega conforme as figuras 24 e 25.

Figura 24 - Tela de Login e Entregas pendentes do Entregador



Fonte: O autor

Figura 25 - Telas de trajeto e registro de ocorrência



Fonte: O autor

4.4. IMPLEMENTAÇÃO DO SISTEMA

4.4.1. Método de desenvolvimento

O sistema foi implementado utilizando a metodologia ágil *SCRUM* como modelo de desenvolvimento, realizando entregas de novas funcionalidades a cada sprint até chegar a um produto entregável. A implementação foi dividida nas seguintes *sprints* como mostrado no quadro 13.

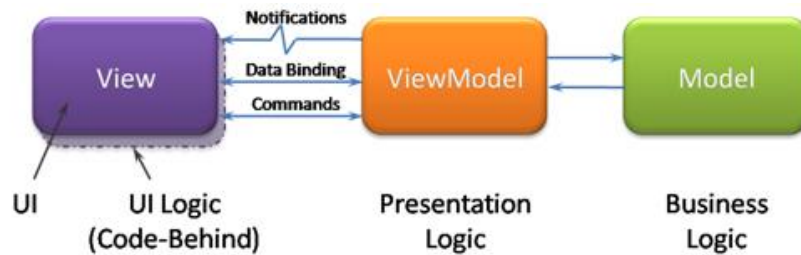
Tabela 13 - Divisão das sprints realizadas ao decorrer do trabalho

Processos	Sprints				
	1ª	2ª	3ª	4ª	5ª
Requisitos	Análise e definição de requisitos básicos.	Revisão de possibilidade de implementação dos requisitos.	-	-	-
Análise	Criação dos diagramas de Caso de uso e de classes.	Ajustes e alterações necessárias no diagrama de Classes.	Revisão da modelagem das classes e casos de uso.	-	-
Projeto	Modelagem do banco de dados.	Ajustes na modelagem.	Revisão da modelagem.	-	-
Implementação	<ol style="list-style-type: none"> 1. Estudo das tecnologias utilizadas. 2. Implementação da funcionalidade de <i>Login</i> da Empresa. 3. Implementação do Modulo de Empresa. 'CRUD'. 	<ol style="list-style-type: none"> 1. Implementação dos "CRUDs" de entregadores e clientes. 2. Implementação dos "CRUDs" de veículos e encomendas. 	<ol style="list-style-type: none"> 1. Integração com banco de dados <i>Firebase</i>. 2. Configurações para utilizar <i>Google Maps API</i>. 3. Implementação das funcionalidades de localizar entregador. 	<ol style="list-style-type: none"> 1. Implementação do Modulo do Cliente (Localização da encomenda) 2. Implementação do Modulo do Cliente (Localização da encomenda) 	<ol style="list-style-type: none"> 1. Revisão e Ajustes no código e layout da aplicação.
Testes	Testes unitários e de caixa preta.	Testes unitários e de caixa preta.	Testes de integração com o banco de dados.	Testes unitários e de integração.	Testes alfa antes da entrega.

4.4.2. Arquitetura do projeto

Para a implementação do aplicativo, foi utilizada a linguagem *Kotlin*, o banco de dados do *Firebase*, e a API do *Google Maps*. Além dessas tecnologias, foram utilizadas a arquitetura MVVM (*Model-View -ViewModel*) conforme a figura 26.

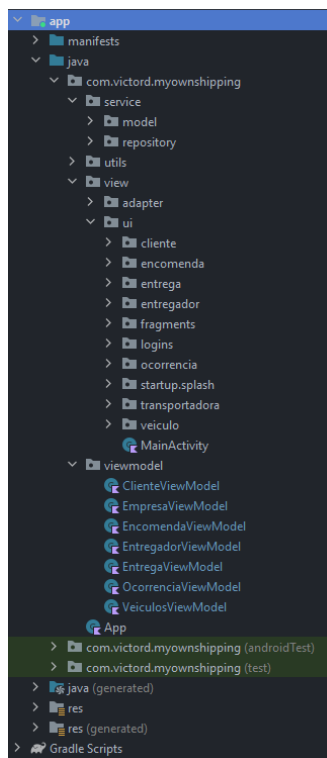
Figura 26 – Arquitetura MVVM



Fonte: (DEV MEDIA)

Utilizando o padrão, foi possível delegar responsabilidades de maneira mais eficiente, deixando cada classe com sua devida função, além de manter uma organização melhor dos pacotes do projeto, conforme a figura 27.

Figura 27 - Projeto aplicando a arquitetura MVVM



Fonte: O autor

Unindo todas essas tecnologias, metodologias e ferramentas foi possível realizar a implementação da aplicação de maneira a seguir boas práticas de programação.

4.4.3. Configurações iniciais

Para iniciar a implementação foi necessário adicionar nosso projeto ao console do *Firebase* e fazer o download do arquivo “google-services.json”, e adicionar ainda o projeto ao *Google Cloud* para consumir as APIs do *Google Maps*, além de adicionar algumas dependências no nosso arquivo *Gradle*, para estar utilizando o *ViewModel*, *Firebase* e as APIs do *Google Maps* como mostrado na figura 28.

Figura 28 - Dependências necessárias no arquivo Gradle.

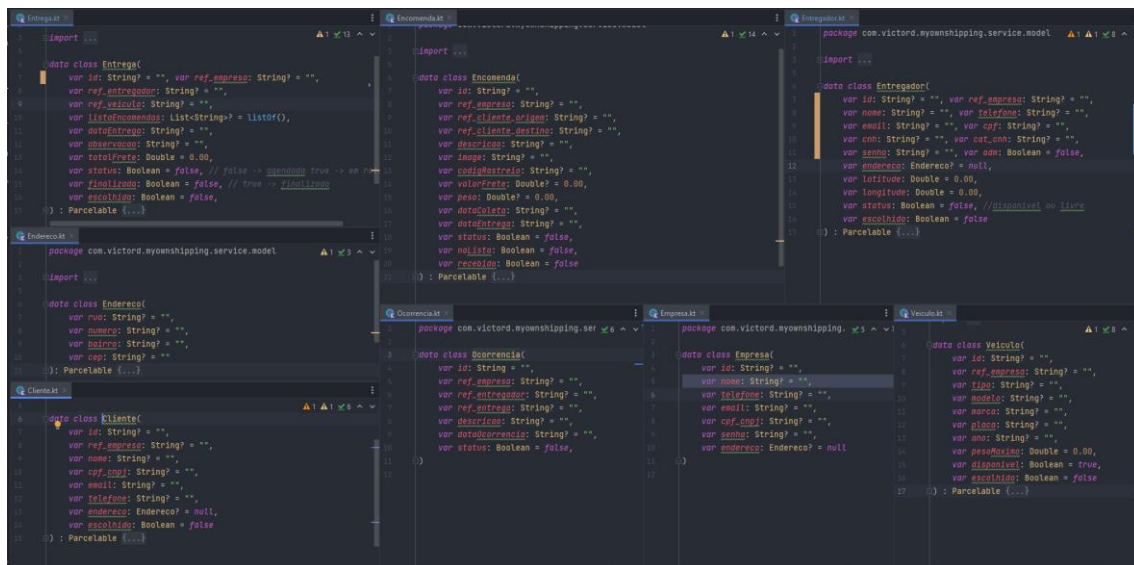
```
dependencies {  
  
    // firebase  
    implementation platform('com.google.firebase:firebase-bom:31.1.0')  
    implementation 'com.google.firebase:firebase-analytics-ktx'  
    implementation 'com.google.firebase:firebase-database-ktx:20.0.4'  
    implementation 'com.google.firebase:firebase-auth-ktx:21.1.0'  
    implementation 'com.google.firebase:firebase-firestore-ktx:24.4.1'  
  
    // google maps  
    implementation 'com.google.android.gms:play-services-maps:18.1.0'  
    implementation 'com.google.android.gms:play-services-location:21.0.1'  
  
    // view model e livedata  
    implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.5.1'  
    implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.5.1'  
}
```

Fonte: O autor

4.4.4. Utilização do MVVM

No que diz respeito ao código da aplicação foi utilizado recursos do *Kotlin* como as *Data Classes* implementando a classe *Parcelable* para realizar os envios de objetos serializados para novas telas representando as classes utilizadas na aplicação conforme a imagem abaixo.

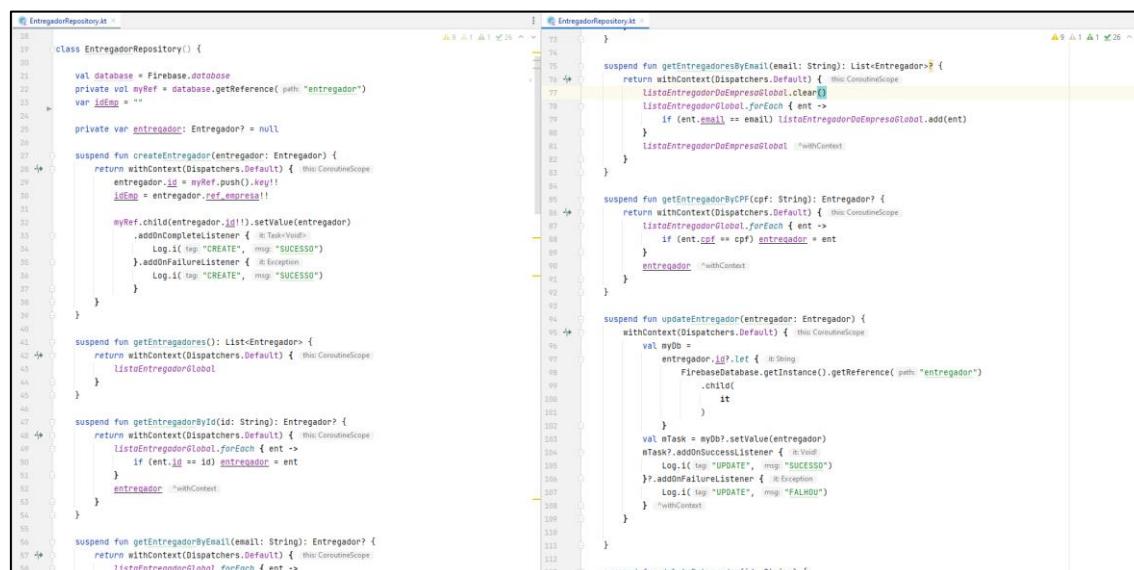
Figura 29 - Código das *data classes* utilizadas



Fonte: O autor

A estrutura utilizada na comunicação entre o banco de dados e as Views foram a criação do pacote *repository* contendo todas as classes repositório, para através dos métodos criados permitir a comunicação com o banco de dados. Funcionando da seguinte maneira, nossas classes servem como moldes para os objetos que estarão sendo salvos e baixados do nosso banco, a criação de classes *repository* servem para abstrair essa lógica que irá ser utilizada pelas nossas *ViewModels* como o que foi feito com a parte de entregadores na figura 30.

Figura 30 - Classe *EntregadorRepository*



```

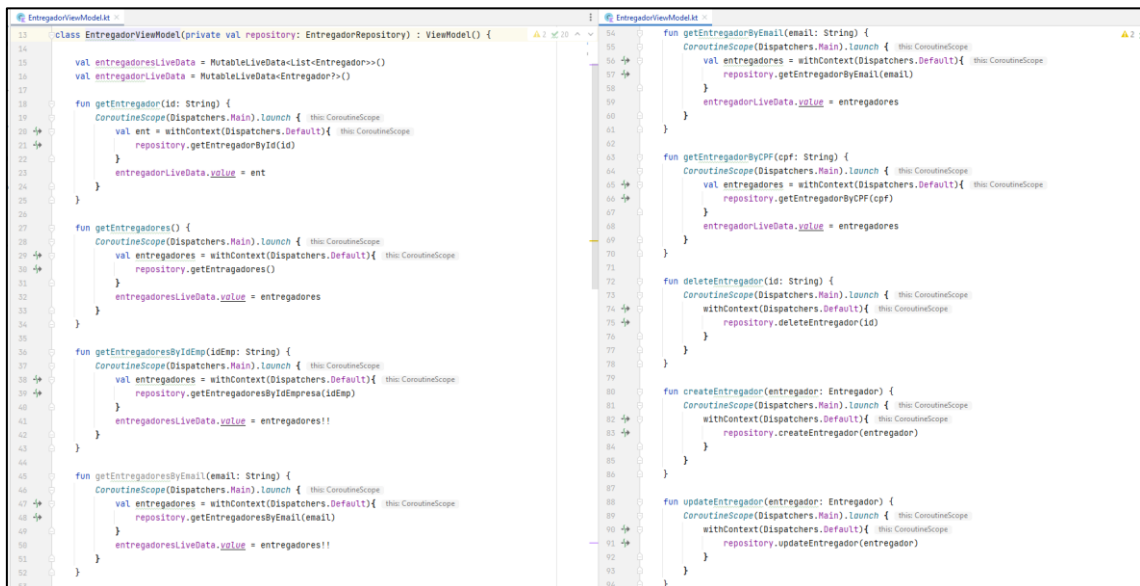
112
113 suspend fun deleteEntregador(id: String) {
114     withContext(Dispatchers.Default) { this: CoroutineScope
115
116         val myDb = myRef.child(id)
117         val mTask = myDb.removeValue()
118
119         mTask.addOnSuccessListener { it: Void!
120             Log.i( tag: "DELETE", msg: "SUCESSO")
121         }.addOnFailureListener { it: Exception
122             Log.i( tag: "DELETE", msg: "FALHOU")
123         }
124     }
125 }
126
127 init {
128     myRef.addValueEventListener(object : ValueEventListener {
129         override fun onDataChange(snapshot: DataSnapshot) {
130
131             val entregadores = snapshot.children.map { dataSnapshot ->
132                 dataSnapshot.getValue(Entregador::class.java)
133             } as MutableList<Entregador>
134
135             listaEntregadorGlobal.clear()
136             if (listaEntregadorGlobal.isEmpty()) {
137                 listaEntregadorGlobal.addAll(entregadores)
138             }
139         }
140
141         override fun onCancelled(error: DatabaseError) {
142             Log.w( tag: "ERRO", msg: "VALORES CANCELADOS")
143         }
144     })
145 }
146
147 }

```

Fonte: O autor

Com a classe criada podemos criar nossa *ViewModel* que vai pôr as informações para a nossa *view*, conforme a figura 31.

Figura 31 - Classe EntregadorViewModel



```

class EntregadorViewModelFactory(
    private val repository: EntregadorRepository
) : ViewModelProvider.Factory {
    override fun <T : ViewModel> create(modelClass: Class<T>): T {
        return EntregadorViewModel(repository) as T
    }
}
    
```

Fonte: O autor

A partir da estruturação das nossas *ViewModels* foi possível criar as *views* com menos responsabilidades, retirando toda a parte lógica que não tem a ver com ela, como na figura 32.

Figura 32 - Classe EntregadoresFragment

```

class EntregadoresFragment : Fragment() {
    private lateinit var binding: FragmentEntregadoresBinding
    private lateinit var adapter: EntregadorAdapter

    private lateinit var viewModel: EntregadorViewModel

    val PREFS_EMPRESA = "EmpresaPrefs"
    var idEmpresa = ""

    private var listaEntregadores = ListaEntregadorDoEmpresaGlobal

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View {
        // Inflate the layout for this fragment
        binding = FragmentEntregadoresBinding.inflate(inflater, container, attachToParent: false)
        val settings: SharedPreferences = requireContext().getSharedPreferences(PREFS_EMPRESA, 0)
        idEmpresa = settings.getString("idEmpresa", "").toString()

        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState: Bundle?) {
        super.onViewCreated(view, savedInstanceState)
        coletarLista()
    }

    override fun onStart() {
        super.onStart()
        coletarLista()
    }

    override fun onResume() {
        super.onResume()
        coletarLista()
        setarListeners()
    }

    private fun coletarLista() {
        viewModel = ViewModelProvider(
            this, EntregadorViewModelFactory(
                EntregadorRepository()
            )
        )[EntregadorViewModel::class.java]

        viewModel.entregadoresLiveData.observe(viewLifecycleOwner, observe { entregadores ->
            listaEntregadores = entregadores.toMutableList()
            gerarViewDaLista()
        })

        viewModel.getEntregadoresByIdEmp(idEmpresa)
    }

    private fun setarListeners() {
        binding.rvEntregadores.setOnRefreshListener {
            coletarLista()
            binding.rvEntregadores.isRefreshing = false
        }

        binding.fabAddEntregador.setOnClickListener { @View()
            val intent = Intent(context, CadastrarEntregadorActivity::class.java)
            startActivity(intent)
            onPause()
        }
    }

    // passar as informações da lista para o adapter
    private fun gerarViewDaLista() {
        adapter = this.requireContext().let { @Context()
            EntregadorAdapter(
                listaEntregadores,
                it, ::onVisualizarEntregador, ::onEditItem
            )
        }
        binding.rvEntregadores.layoutManager = LinearLayoutManager(context)
        binding.rvEntregadores.adapter = adapter
    }

    // quando o nome do entregador é clicado essa função é executada
    fun onVisualizarEntregador(entregador: Entregador) {
        val position = listaEntregadorGlobal.indexOf(entregador)
        val intent = Intent(context, VisualizarEntregadorActivity::class.java)
        intent.putExtra( name: "idEntregador", listaEntregadorGlobal[position].id)
        startActivity(intent)
    }

    fun onEditItem(entregador: Entregador) {
        val position = listaEntregadorGlobal.indexOf(entregador)
        val intent = Intent(context, EditarEntregadorActivity::class.java)
        intent.putExtra( name: "idEntregador", listaEntregadorGlobal[position].id)
        startActivity(intent)
    }
}

```

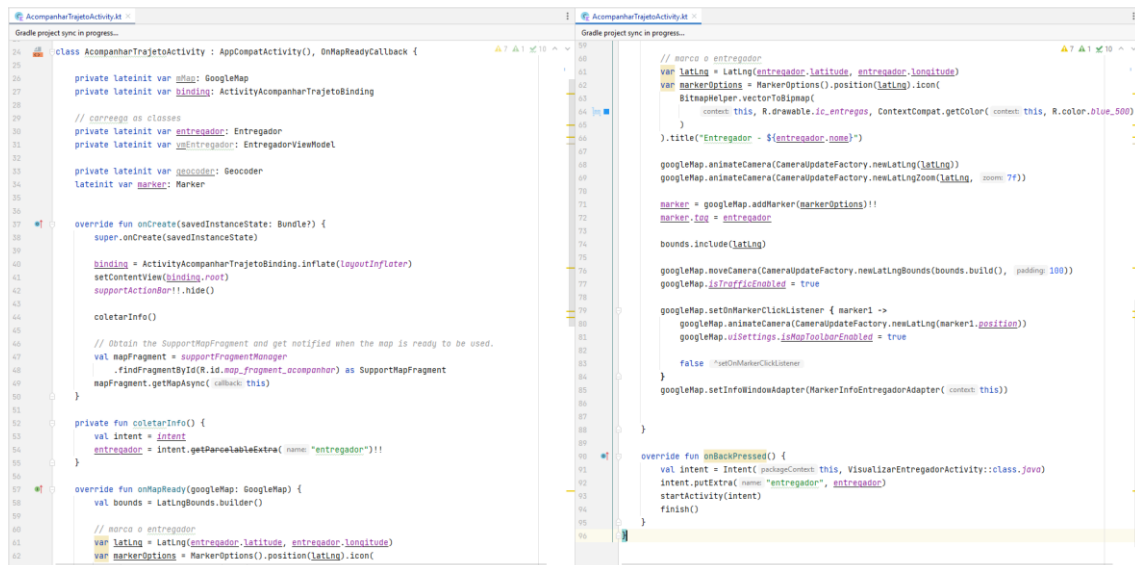
Fonte: O autor

Esta mesma abordagem foi seguida para as demais classes que precisavam ter contato com o banco de dados.

4.4.5. Buscando a localização com as APIs do Google Maps

Nas classes que precisavam obter localização por meio da API, foi utilizado *activities* pré-moldadas para a geração de mapas, depois foi colocada nossa lógica para obtenção da localização exata no mapa conforme a figura 33.

Figura 33 - Classe AcompanharTrajetoActivity



Fonte: O autor

A classe implementa a classe *OnMapReadyCallback* para utilizarmos a geração do mapa na construção da *view*. E com essas exemplificações seguindo sempre a mesma estrutura, tentando sempre seguir as boas práticas que a aplicação foi implementada.

5 CONSIDERAÇÕES FINAIS E DIRECIONAMENTO PARA TRABALHOS FUTUROS

5.1. INTRODUÇÃO

No presente trabalho, buscou-se basear em aplicações e situações existentes e que já atuam ou atuaram no mercado ou se fazem presentes no dia a dia, desta maneira pensou-se em requisitos e funcionalidades que pudessem suprir ou auxiliar no gerenciamento e monitoramento de entregas conforme proposto no escopo do trabalho. Os objetivos específicos definidos foram em sua forma básica alcançados.

5.2. CONCLUSÃO

A aplicação implementada pode ser empregada em pequenas empresas que necessitem de uma ferramenta para gerenciar as suas entregas e precisam de uma forma de monitorá-las, com a facilidade de ter o aplicativo precisando apenas ter um dispositivo mobile, para baixar e utilizar a ferramenta de auxílio nessas atividades.

As funcionalidades básicas foram implementadas, como cadastro de entregas, entregadores, clientes, encomendas, veículos e etc., e suas outras operações básicas. Além das funcionalidades de localizar encomenda, traçar rotas e confirmar entregas. Pode-se dizer que com o que foi proposto o aplicativo cumpre mesmo que de maneira simples seus requisitos, ainda podendo melhorar em suas funcionalidades.

5.3. TRABALHOS FUTUROS

De modo geral, a aplicação abre portas para a sua melhoria e incremento de forma a tornar-se uma ferramenta importante no auxílio de suas atividades. Pensando nisso, pode-se buscar a implementação de melhorias e novas soluções de acordo com as necessidades que precisem ser atendidas, além de permitir novas possibilidades.

6 REFERÊNCIAS

BALLOU, Ronald H. **Gerenciamento da Cadeia de Suprimentos-: Logística Empresarial**. Bookman editora, 2009.

DA SILVA, Diana Aires; MOREIRA, Me Ronaldo Emerick. **O E-commerce como Estratégia no Processo de Expansão dos Negócios de Pequenas Empresas**. Revista de Administração do UNIFATEA, v. 3, n. 3, 2010.

GERHARDT, Tatiana Engel; SILVEIRA, Denise Tolfo. **Métodos de pesquisa**. Plageder, 2009.

GOOGLE DEVELOPERS. **Conheça o Android Studio**. Disponível em: <<https://developer.android.com/studio/intro>>. Acesso em: 14 de junho de 2022.

FERREIRA, Karine Araújo; RIBEIRO, Priscilla Cristina Cabral. **Tecnologia da Informação e Logística: os impactos do EDI nas operações logísticas de uma empresa do setor automobilístico**. XXIII ENEGEP-Encontro Nacional de Engenharia de Produção, Ouro Preto, 2003.

IBGE. **Censo Agro 2017**. Disponível em: <<https://censoagro2017.ibge.gov.br/2013-agencia-de-noticias/releases/28668-ibge-divulga-estimativa-da-populacao-dos-municipios-para-2020.html#:~:text=O%20munic%C3%ADpio%20de%20S%C3%A3o%20Paulo>>. Acesso em: 17 jun. 2022.

KOTESKI, Marcos Antonio. **As micro e pequenas empresas no contexto econômico brasileiro**. Revista FAE Business, v. 8, n. 1, p. 16-18, 2004.

LAUDON, K. C. LAUDON. JP. **Sistemas de Informação Gerenciais**. 2004.

LECHETA, Ricardo R. **Google Android-3ª Edição: Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. Novatec Editora, 2013.

MOURA, Benjamim. **Logística: conceitos e tendências**. Centro Atlantico, 2006.

NOVAES, Antonio. **Logística e gerenciamento da cadeia de distribuição – 3ª Edição**. Elsevier Brasil, 2007.

SALES, Rodrigo Lacerda; DE BARROS, Aloisio Antonio; DE ARAÚJO PEREIRA, Cláudia Maria Miranda. **Fatores condicionantes da mortalidade dos pequenos negócios em um típico município interiorano brasileiro**. Revista da micro e pequena empresa, v. 2, n. 2, p. 38-55, 2008.

SCHARF, Danrley. **Aplicação Web para gerenciamento de entregas de transportadora**. 2018. 50f. TCC - Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas, Universidade Tecnológica Federal do Paraná,

Campus Pato Branco. Pato Branco, 2018. Disponível em:
http://repositorio.utfpr.edu.br/jspui/bitstream/1/15534/1/PB_COADS_2018_2_03.pdf. Acesso em: 14 de junho de 2022.

SCHWABER, Ken; SUTHERLAND, Jeff. **The 2020 SCRUM guide**. Retrieved May, v. 6, p. 2021, 2020.

CSCMP. **SCM Definitions and Glossary of Terms**. Disponível em:
<https://cscmp.org/CSCMP/Academia/SCM_Definitions_and_Glossary_of_Terms/CSCMP/Educate/SCM_Definitions_and_Glossary_of_Terms.aspx?hkey=60879588-f65f-4ab5-8c4b-6878815ef921>. Acesso em: 14 de junho de 2022.

WEIBER, Felipe. **Sistema de Gerenciamento de Pizzaria**. 2017. 28 f. TCC - Curso de Tecnologia em Sistemas para Internet, Universidade Tecnológica Federal do Paraná. Guarapuva, 2017. Disponível em:
http://repositorio.utfpr.edu.br/jspui/bitstream/1/11853/2/GP_COINT_2017_2_01.pdf. Acesso em: 14 de junho de 2022.