

EDUARDO FEITOSA COSTA - 1370481923016

THIAGO MARCELLO AMORIM OLIVEIRA - 1370482013008

VICTOR LUÍS MARANHÃO LELLIS - 1370482013026

**APLICATIVO PARA COLETA DE INFORMAÇÕES, VOLTADO PARA PROMOTORES DE MERCHANDISING**

São Paulo

2022

EDUARDO FEITOSA COSTA - 1370481923016

THIAGO MARCELLO AMORIM OLIVEIRA - 1370482013008

VICTOR LUÍS MARANHÃO LELLIS - 1370482013026

**APLICATIVO PARA COLETA DE INFORMAÇÕES VOLTADO PARA PROMOTORES DE MERCHANDISING**

Trabalho de Conclusão de Curso apresentado à Faculdade de Tecnologia da Zona Sul – Fatec-ZS, como exigência parcial para a obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof.Esp.Eduardo Bonetti de Freitas.

São Paulo

2022

**BANCA EXAMINADORA**

Professor Orientador

Prof. Esp.Eduardo Bonetti de Freitas \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professor

Prof. Dr.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Professor

Prof. Ms. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**DEDICATÓRIA**

Dedicamos este trabalho aos professores da Fatec Zona Sul e às nossas famílias.

**AGRADECIMENTOS**

Agradecemos ao professor Eduardo Bonetti de Freitas pela dedicação e orientação dada; e aos professores, em geral, da Fatec Zona Sul, por nos ensinar tudo o que utilizamos para produzir este trabalho.

**RESUMO**

Este trabalho de conclusão do curso superior tecnólogo de Análise e Desenvolvimento de Sistemas da FATEC Zona Sul - Dom Paulo Evaristo Arns, desenvolvemos um aplicativo mobile utilizando todas as habilidades desenvolvidas ao longo do curso, junto da documentação do mesmo, visando auxiliar promotores de merchandising de diferentes seguimentos a cumprirem seus roteiros e objetivos de operação, quanto as coletas de informações solicitadas pelo cliente.

O projeto foi construído principalmente em *Kotlin,* que hoje é a linguagem padrão e recomendada pela google nas construções de aplicativos. Também foi utilizado a linguagem Java no desenvolvimento do aplicativo, PHP para troca de informações com o banco de dados, onde utilizamos *SQL Server Management* Studio para o gerenciamento dos dados.

**Palavras-Chave:** Banco de dados, Desenvolvimento de software, *Android*, SQL, Aplicativo *Mobile*.

***ABSTRACT***

*Monography required to complete the bachelor’s degree in system Analysis and Development at the FATEC Zona Sul - Dom Paulo Evaristo Arns, using all the skills developed throughout the course for the development of a software, along with its documentation, in order to help merchandising promoters from different segments to fulfill their itineraries and operation objectives, as well as the collection of information requested by the client.*

*The project was built mainly in Kotlin, which is now the default language and recommended by google for building applications. The Java language was also used in the development of the app, PHP to exchange information with the database, where we use SQL Server Management Studio to manage the data.*

***Keywords:*** *Database, Software development, Android, SQL, Mobile Application*.

**LISTA DE FIGURAS**

[**Figura 1** - Cronograma de desenvolvimento. 17](#_Toc109483064)

[**Figura 2** - Matriz de entrevista. 20](#_Toc109483065)

[**Figura 3** - Diagrama de caso de uso. 24](#_Toc109483066)

[**Figura 4** - Modelo lógico. 28](#_Toc109483067)

[**Figura 5** – Tela de login. 31](#_Toc109483068)

[**Figura 6** - Tela de início. 32](#_Toc109483069)

[**Figura 7** - Tela de perfil. 32](#_Toc109483070)

[**Figura 8** - Tela de roteiro. 33](#_Toc109483071)

[**Figura 9** - Tela de coleta. 34](#_Toc109483072)

[**Figura 10** - Coleta de produto. 34](#_Toc109483073)

[**Figura 11** - Diálogo produto. 35](#_Toc109483074)

[**Figura 12** - Tela de coleta com indicadores. 36](#_Toc109483075)

[**Figura 13** - Tela de roteiro com indicadores. 36](#_Toc109483076)

**LISTA DE TABELAS**

[**Tabela 1** – Tabela de pessoa. 24](#_Toc109482265)

[**Tabela 2** – Tabela de perfil. 24](#_Toc109482266)

[**Tabela 3** – Tabela de bandeira. 25](#_Toc109482267)

[**Tabela 4** - Tabela de loja. 25](#_Toc109482268)

[**Tabela 5** - Tabela de sku. 25](#_Toc109482269)

[**Tabela 6** – Tabela de marca. 25](#_Toc109482270)

[**Tabela 7** – Tabela de vínculo produto bandeira. 25](#_Toc109482271)

[**Tabela 8** – Tabela de roteiro. 26](#_Toc109482272)

[**Tabela 9** – Tabela de justificativa. 26](#_Toc109482273)

[**Tabela 10** – Tabela de mobile pesquisa. 26](#_Toc109482274)

[**Tabela 11** – Tabela de mobile pesquisa produto. 26](#_Toc109482275)

[**Tabela 12** – Tabela de relacionamentos. 26](#_Toc109482276)

[**Tabela 13** – Tabela de detalhamento de relacionamentos. 27](#_Toc109482277)

**SUMÁRIO**

[1 INTRODUÇÃO 10](#_Toc109739167)

[1.1 TEMA 11](#_Toc109739168)

[1.2 JUSTIFICATIVA 11](#_Toc109739169)

[1.3 REFERENCIAL TEÓRICO 11](#_Toc109739170)

[1.3.1 Logística 11](#_Toc109739171)

[1.3.2 Desenvolvimento de aplicações móveis 12](#_Toc109739172)

[1.4 DELIMITAÇÃO DO PROBLEMA 13](#_Toc109739173)

[1.5 FORMULAÇÕES E HIPÓTESES 13](#_Toc109739174)

[1.6 TÍTULO 14](#_Toc109739175)

[1.7 OBJETIVOS 14](#_Toc109739176)

[1.8 FERRAMENTaS DE DESENVOLVIMENTO 14](#_Toc109739177)

[1.8.1 Biblioteca e *frameworks*: 14](#_Toc109739178)

[1.8.2 Plataformas de desenvolvimento: 15](#_Toc109739179)

[1.8.3 Linguagens de desenvolvimento: 15](#_Toc109739180)

[1.9 CRONOGRAMA DO PROJETO 17](#_Toc109739181)

[2 ENTREVISTA 18](#_Toc109739182)

[2.1 OBJETIVO GERAL 18](#_Toc109739183)

[2.1.1 Objetivos específicos 18](#_Toc109739184)

[2.1.2 Roteiro de entrevista: 18](#_Toc109739185)

[2.2 DESCRIÇÃO DA NECESSIDADE DO CLIENTE 21](#_Toc109739186)

[2.3 REQUISITOS FUNCIONAIS 21](#_Toc109739187)

[2.4 REQUISITOS NÃO-FUNCIONAIS 22](#_Toc109739188)

[3 DESENHO DO SISTEMA 23](#_Toc109739189)

[3.3 Dicionário de dados 24](#_Toc109739190)

[4 ARQUITETURA DO SISTEMA 29](#_Toc109739191)

[4.1 PROJETO DE INTERFACE DO SISTEMA 29](#_Toc109739192)

[4.1.1 Layouts de telas e descrição 30](#_Toc109739193)

[7CONCLUSÃO (Numerar ítem 7) Refazer a Conclusão....o TCC não apresenta um desfecho conclusivo) Utilizem uma página com aprox.3 paragrafos para desenvolver a conclusão. 37](#_Toc109739194)

[REFERÊNCIAS BIBLIOGRÁFICAS 38](#_Toc109739195)

# INTRODUÇÃO

Pela definição do *Council of Supply Chain Management Professionals*,

Logística é a parte do Gerenciamento da Cadeia de Abastecimento que planeja, implementa e controla o fluxo e armazenamento eficiente e econômico de matérias-primas, materiais semi-acabados e produtos acabados, bem como as informações a eles relativas, desde o ponto de origem até o ponto de consumo, com o propósito de atender às exigências dos clientes. (CARVALHO, 2002, p. 31)

Com a escalabilidade dos dados e a queda no custo de armazenamento, nas últimas décadas, houve um volume exponencial enorme de dados, a todo instante, nas organizações (REZENDE, 2005). O principal objetivo de manter os dados armazenados, segundo Camilo e Silva (2009), é auxiliar os tomadores de decisões das empresas em suas estratégias. Porém, com o aumento do volume dos dados ficou difícil utilizar as técnicas tradicionais para tratar os dados, tornando estas empresas, “ricas em dados, mas pobres em informação” (HAN; KAMBER; PEI, 2012).

 Com a necessidade da evolução dos processos de análise de dados, no fim da década de oitenta surgiu o conceito de Descoberta de Conhecimento, onde o foco principal é a extração do conhecimento por meio de grandes bases de dados (REZENDE, 2005).

Portanto, considerando a ampla adoção dos celulares no Brasil, com o uso corporativo, o objetivo deste trabalho é desenvolver um aplicativo móvel para ser utilizado na coleta de dados dos produtos que ficam expostos nos estabelecimentos comerciais, e utilizar esses dados para analisar as variações dos preços, localização dos produtos nas prateleiras e promover ações estratégicas para melhorar as vendas.

## TEMA

A utilização de aplicativo móvel para coletar e verificar as informações dos produtos nas prateleiras dos estabelecimentos comerciais com o objetivo de padronizar os dados e utilizá-los, de forma ágil, em ações estratégicas para otimizar ou alavancar as vendas dos produtos que não estão gerando lucros satisfatórios.

## JUSTIFICATIVA

A melhoria do serviço é uma forma eficaz da organização se firmar no mercado, através da conquista do cliente satisfeito. Nickels e Wood (1999) relatam que o serviço ao cliente é o conjunto de atividades desenvolvidas pela empresa na busca da satisfação e fidelização dos clientes, proporcionando, ao mesmo tempo, uma percepção de que a organização pode ser um ótimo parceiro comercial. Este relacionamento faz com que ambas as partes se beneficiem na negociação "ganha-ganha".

Este trabalho busca trazer melhorias comerciais para as marcas que fazem a distribuição de seus produtos em supermercados; essa aplicação proporcionará, às marcas, coletar dados, rapidamente, sobre onde seus produtos estão posicionados, e o preço que estão sendo comercializados em cada estabelecimento; essas informações têm como objetivo transformar em insights e serem utilizados de forma estratégica para potencializar as vendas e projetar a reposição do estoque dos clientes, para que não haja conflitos e atrasos no abastecimento dos produtos.

## REFERENCIAL TEÓRICO

### Logística

De acordo com o autor Ching (2001), a evolução da logística foi promovida por inúmeras contribuições, dentre elas algumas econômicas e tecnológicas. Segundo ele, de início, no período pós-guerra, houve mudanças no relacionamento com o consumidor, na qual houve alterações nas distâncias a serem atendidas, e pela procura de uma maior variedade de mercadorias/serviços; depois disso, foi possível calcular os custos dos serviços logísticos. A partir de 1990, foram desenvolvidos conceitos e princípios logísticos conforme às necessidades das empresas, com o objetivo de melhorar as entregas e custos dos serviços, deixando as empresas mais competitivas.

Para Fernandes (2012), a logística refere-se “[...] à responsabilidade de construir e gerir sistemas para controlar o transporte e a localização dos estoques de materiais, e obter ou proporcionar insumos e produtos inacabados pelo menor custo total”. Com isso, as empresas devem coordenar suas atividades de acordo com uma estratégia coerente, considerando a natureza global de seus negócios.

Outro autor, Taboada (2012), ele diz que a globalização tem transformado rapidamente a forma de organizar e administrar os negócios, em vista que as barreiras de espaço, tempo e localização têm sido removidas; ou seja, não é mais uma limitação para quem vende e para quem compra produtos e serviços.

Portanto, a logística é fundamental no processo de decisão da estratégia organizacional e não ser apenas uma área que executa decisões tomadas anteriormente por outros. Portanto, cada vez mais empresas vêm entendendo que a logística consiste em um departamento estratégico e os materiais se desloquem de forma otimizada para que a capacidade e a complexidade das operações logísticas sejam mais eficientes e lucrativas (FERNANDES, 2012).

### Desenvolvimento de aplicações móveis

Atualmente, o software se tornou vital para a escalabilidade e lucratividade das organizações, pois se torna “um método de captar dados e/ou conhecimento, um modo de diferenciar os produtos de seus competidores e uma forma para o conhecimento coletivo de uma organização” (PRESSMAN, 2002, p.828); portanto, quanto mais propício as mudanças estiverem ao usuário final, sua utilização apresenta mais segurança, conforto e eficiência do sistema informatizado. Em qualquer trabalho, o software é o fator que diferencia e proporciona vantagem competitiva.

A utilização dos dispositivos móveis (celulares) na última década ao redor do mundo fez com que o desenvolvimento de aplicações móveis se tornasse essencial para a praticidade e comodidade da atualidade. Há diversos sistemas operacionais (SO) para os dispositivos existentes atualmente, e para cada tipo de sistema operacional existem linguagens de programação específicas para serem utilizadas na construção dos aplicativos, como, por exemplo, o *Java* e o *Kotlin* para a plataforma *Android*, e o *Objective-C* para a plataforma iOS, sendo essas duas as plataformas mais frequentemente usadas entre os dispositivos (Goasduff, 2012).

As aplicações desenvolvidas a partir dessas linguagens específicas são chamadas de nativas, pois conseguem, através das *Application Programming Interfaces* (API) disponibilizadas, acesso aos componentes de *hardware* do dispositivo, como câmera, geolocalização e microfone (white, 2013). Os aplicativos nativos são distribuídos pelas próprias empresas que cuidam de cada um dos sistemas operacionais através das suas lojas de aplicativos, podendo haver uma verificação daqueles antes de sua publicação dentro das lojas (WHITE, 2013). Por outro lado, existem aplicativos não nativos, que são projetados para serem compatíveis em múltiplos sistemas operacionais, os quais são construídos com apenas um único código, geralmente escrito em HTML, CSS e *Javascript* (MARTINS et al, 2013). No entanto, uma das principais desvantagens ao construir-se uma aplicação híbrida é o fato de os recursos nativos do dispositivo não poderem ser acessados diretamente através da linguagem utilizada. Os aplicativos não nativos podem acessar alguns desses recursos, como a câmera ou a localização do usuário, mas o fazem usando métodos não ideais, nos diz Martins (2013).

Com o avanço da tecnologia, foram surgindo diversas bibliotecas e *frameworks* a fim de facilitar o processo de desenvolvimento de aplicativos móveis híbridos, tornando o processo mais produtivo. Atualmente, há uma enorme gama de *frameworks* específicos para cada tipo de necessidade.

## DELIMITAÇÃO DO PROBLEMA

Como a utilização de uma aplicação móvel poderá colaborar com a sustentabilidade e com as empresas, no sentido de coletar as informações rapidamente e transformá-las e ações estratégicas.

## FORMULAÇÕES E HIPÓTESES

* A tecnologia é utilizada como processo de melhoria e otimização nos processos logísticos.
* Os dados coletados através do aplicativo oferecem uma maior confiabilidade das informações.
* A utilização de ferramentas tecnológicas proporcionará ações estratégicas para melhorar as vendas das marcas nos estabelecimentos comerciais.
* As informações dos produtos nos estabelecimentos resultarão em processos de melhorias para alavancar o faturamento.

## TÍTULO

Aplicativo para coleta de informações, voltado para promotores de *merchandising*.

## OBJETIVOS

* Proporcionar agilidade na coleta de informações;
* Verificar por meio de análises quais os melhores layouts que melhoram a venda dos produtos;
* Quais os preços praticados pelos revendedores;
* Transformar os dados em ações estratégicas;

## FERRAMENTaS DE DESENVOLVIMENTO

### Biblioteca e *frameworks*:

* *AndroidX;* é uma grande melhoria para a Biblioteca de Suporte original do Android, que não é mais mantida. Os pacotes do *androidx* substituem totalmente a Biblioteca de Suporte, fornecendo paridade de recursos e novas bibliotecas.
* *Room;* é uma das bibliotecas existentes dentro do conjunto *“Android JetPack”* apresentado durante o Google I/O de 2018, ela auxilia os desenvolvedores criando uma abstração das camadas de banco de dados *(SQLite)*.
* *Retrofit*; é uma API desenvolvida pela Square seguindo padrão *REST*, fornecendo um padrão simples de implementação para transmissão de dados entre aplicação e servidor, que faz uso do *JSON*. Este artigo tem por finalidade a apresentação da API *Retrofit* de conexão *HTTP* para *Android*.
* *Fragment;* representa o comportamento ou uma parte da *interface* do usuário em um *FragmentActivity*. É possível combinar vários fragmentos em uma única atividade para criar uma IU de vários painéis e reutilizar um fragmento em diversas atividades. (uma espécie de “subatividade” que pode ser reutilizada em diferentes atividades).
* *Data Binding Library;* é uma biblioteca de apoio que permite vincular componentes de IU dos seus layouts a fontes de dados do aplicativo usando um formato declarativo, em vez de programático.
* *ConstraintLayout;* oferece a capacidade de projetar completamente sua *interface* do usuário com o recurso de arrastar e soltar fornecido pelo editor de design do *Android Studio*. Isso ajuda a melhorar o desempenho da *interface* do usuário em relação a outros *layouts*. Com a ajuda do *ConstraintLayout,* podemos controlar o grupo de *widgets* através de uma única linha de código. Com a ajuda do *ConstraintLayout,* podemos facilmente adicionar animações aos componentes da *interface* do usuário que usamos em nosso aplicativo.
* *O RecyclerView*; é uma nova *view* que veio para substituir o *ListView* e o *GridView.* De acordo com sua documentação, trata-se de um *widget* mais avançado e eficiente, quando comparado aos seus antecessores, e que apresenta diversas simplificações para suportar animações e diferentes disposições de elementos.
* *Glide*; é uma biblioteca *Image Loader* para *Android* desenvolvida pela *bumptech* e é uma biblioteca recomendada pelo Google. Ele fornece suporte a GIF animado e lida com o carregamento / armazenamento de imagens em cache. O suporte para GIF animado não está implementado no Picasso

### Plataformas de desenvolvimento:

* *Android Studio*: Ambiente de Desenvolvimento Integrado (ou IDE, sigla em inglês para *Integrated Development Environment)*, um programa de computador que reúne as características e ferramentas de apoio para a criação de aplicativos para dispositivos móveis para *Android*.
* *Visual Studio Code*: Editor de código-fonte autônomo que é executado no *Windows, macOS e Linux*. A melhor escolha para desenvolvedores *JavaScript e Web*, com toneladas de extensões dar suporte a praticamente qualquer linguagem de programação.
* *SQL Server Management Studio: O SSMS (SQL Server Management Studio)* é um ambiente integrado para gerenciar qualquer infraestrutura de SQL, do SQL *Server* para o Banco de Dados SQL do Azure. O SSMS fornece ferramentas para configurar, monitorar e administrar instâncias do SQL *Server* e bancos de dados. Use o SSMS para implantar, monitorar e atualizar os componentes da camada de dados usados pelos seus aplicativos, além de criar consultas e *scripts.*

### Linguagens de desenvolvimento:

* *Kotlin;* é uma linguagem de programação *open source* multiplataforma criada pela *JetBrains* e amplamente usada por desenvolvedores *Android*. É multiparadigma, sendo totalmente orientada a objetos, mas com algumas características de linguagem funcional (como as funções *lambda*), com tipagem estática e executada pela *Java Virtual Machine (Máquina Virtual do Java).*
* *Java:* é linguagem de programação orientada a objetos (OPP) e baseado em classes. Seu foco é a segurança, simplicidade, portabilidade e alta performance dos programas. Ao contrário do que ocorre em outras linguagens de programação, as aplicações desenvolvidas em Java não são compiladas em um código nativo para ser executado diretamente pelo dispositivo. Nela, existe a compilação para um código intermediário, o *bytecode*, que é interpretado e executado pela Máquina *Virtual Java (JVM).* Assim, qualquer plataforma que possua a JVM instalada pode executar as aplicações em Java. Daí o fato de essa linguagem ser portátil. Além disso, vale dizer que essa linguagem possui uma sintaxe similar a C e C++, porém em um modelo mais simples, uma extensa biblioteca de rotinas e APIs e um gerenciamento automático de memória, o *Garbage Collector.*
* PHP: O PHP (um acrônimo recursivo para PHP: *Hypertext Preprocessor*) é uma linguagem de *script open source* de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

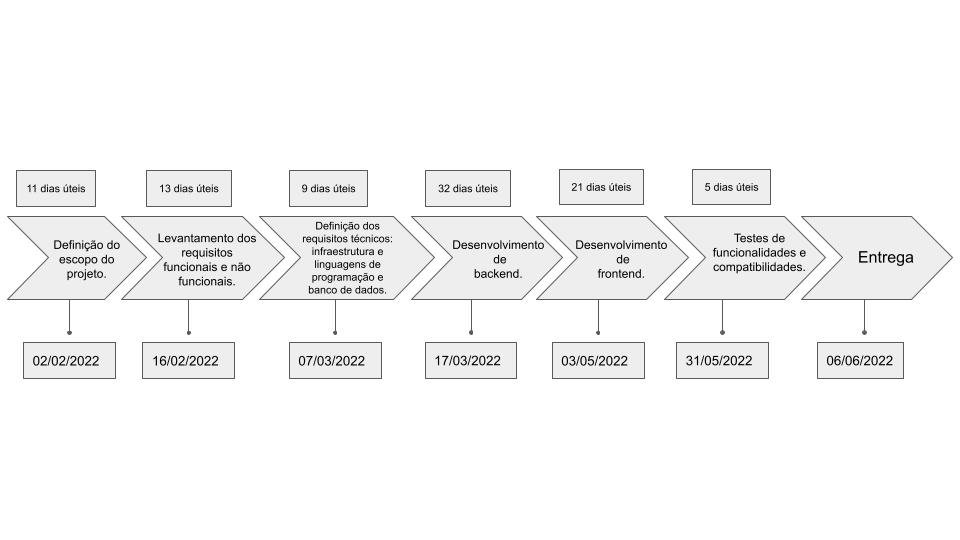
O que distingue o PHP de algo como o *JavaScript* no lado do cliente é que o código é executado no servidor, gerando o HTML que é então enviado para o navegador. O navegador recebe os resultados da execução desse *script*, mas não sabe qual era o código fonte. Você pode inclusive configurar seu servidor web para processar todos os seus arquivos HTML com o PHP, e então não há como os usuários dizerem o que você tem na sua manga.

## CRONOGRAMA DO PROJETO

Segundo Silveira (2010), o cronograma deve conter as atividades principais que dimensionem o tempo necessário para a realização de cada atividade no planejamento da pesquisa: revisão de literatura, coleta de dados, análise dos dados, entre outros.

O cronograma foi elaborado conforme a reunião feita pelos integrantes visando colocar datas com margem para incidentes no percurso do desenvolvimento do aplicativo.

**Figura 1** - Cronograma de desenvolvimento.



Fonte: De autoria própria.

# ENTREVISTA

As entrevistas foram realizadas no período de janeiro e fevereiro de 2022 através do formulário do Google Forms, no qual buscamos profissionais com o cargo de promotor no LinkdIn para responder a entrevista e enviar o formulário nos grupos de trabalho para gerar um maior alcance de pessoas e respostas.

## OBJETIVO GERAL

Perceber as necessidades dos promotores e pesquisadores (usuários do aplicativo) na execução do seu trabalho.

### Objetivos específicos

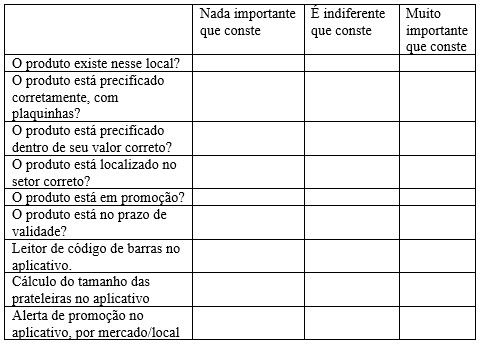
* Identificar a percepção do usuário da necessidade, ou não, do aplicativo, e como isso afetaria seu trabalho;
* Identificar as necessidades em relação ao cadastro do roteiro; ao cadastro dos dados dos produtos, das frentes, gôndolas; e ao cadastro das quantidades de produtos no estoque;
* Estimar a acessibilidade desse grupo aos smartphones e Internet (3G, 4G e 5G).

### Roteiro de entrevista:

Entrevista a ser realizada com o cliente que encomendou o desenvolvimento do aplicativo, promotores e pesquisadores. A entrevista é do tipo estruturada que, conforme Minayo (2009), consta de perguntas fechadas, no caso com alternativas, para compreender e identificar, minuciosamente, as necessidades do cliente.

* Você é:
* Proprietário/gerente.
* Promotor.
* Você já utilizou, ou utiliza, algum aplicativo móvel no seu trabalho?
* Nunca utilizou.
* Utiliza esporadicamente, para algumas tarefas específicas.
* Utiliza constantemente, para praticamente todas as tarefas realizadas.
* Aproximadamente, qual a porcentagem de tempo, com relação ao seu trabalho, que você utiliza algum aplicativo?
* Nunca utiliza.
* Até 20% do seu tempo trabalhado.
* De 20 a 50% do seu tempo trabalhado.
* De 50 a 70% do seu tempo trabalhado.
* Mais de 70% do seu tempo trabalhado.
* Caso você tivesse acesso a um aplicativo móvel no trabalho, qual sua percepção de necessidade deste?
* Totalmente desnecessário.
* Necessário para algumas tarefas específicas.
* Necessário para a maioria das tarefas realizadas.
* No caso de ter a opção de utilizar um aplicativo no seu trabalho, qual sua percepção sobre o quanto o aplicativo lhe ajudaria ou atrapalharia na tarefa de cadastro do roteiro. Marque quantas alternativas se aplicarem a isso:
* Não faria diferença usá-lo ou não.
* Você gastaria mais tempo por usá-lo.
* Você teria ganho de tempo por usá-lo.
* Simplificaria seu trabalho de registro do cadastro.
* Tornaria mais complicado seu trabalho de registro do cadastro.
* No caso de ter a opção de utilizar um aplicativo no seu trabalho, qual sua percepção sobre o quanto o aplicativo lhe ajudaria ou atrapalharia na tarefa de cadastro dos dados dos produtos. Marque quantas alternativas se aplicarem a isso:
* Não faria diferença usá-lo ou não.
* Você gastaria mais tempo por usá-lo.
* Você teria ganho de tempo por usá-lo.
* Simplificaria seu trabalho de registro dos dados dos produtos.
* Tornaria mais complicado seu trabalho de registro dos dados dos produtos.
* No caso de ter a opção de utilizar um aplicativo no seu trabalho, qual sua percepção sobre o quanto o aplicativo lhe ajudaria ou atrapalharia na tarefa de cadastro das frentes e/ou gôndolas. Marque quantas alternativas se aplicarem a isso:
* Não faria diferença usá-lo ou não.
* Você gastaria mais tempo por usá-lo.
* Você teria ganho de tempo por usá-lo.
* Simplificaria seu trabalho de registro das frentes e/ou gôndolas.
* Tornaria mais complicado seu trabalho de registro das frentes e/ou gôndolas.
* No caso de ter a opção de utilizar um aplicativo no seu trabalho, qual sua percepção sobre o quanto o aplicativo lhe ajudaria ou atrapalharia na tarefa de cadastro das quantidades de produtos no estoque. Marque quantas alternativas se aplicarem a isso:
* Não faria diferença usá-lo ou não.
* Você gastaria mais tempo por usá-lo.
* Você teria ganho de tempo por usá-lo.
* Simplificaria seu trabalho de registro das quantidades de produtos no estoque.
* Tornaria mais complicado seu trabalho de registro das quantidades de produtos no estoque.
* Considere cada item a seguir, quanto à *necessidade de constar*, ou não, no aplicativo:

**Figura 2** - Matriz de entrevista.



Fonte: De autoria própria.

* Estimar a acessibilidade desse grupo aos smartphones
* Estimar a acessibilidade desse grupo à Internet (3G, 4G e 5G)

## Descrição da necessidade do cliente

Segundo Kotler (2000), podemos identificar cinco tipos de necessidades do cliente: 1) as declaradas, ou seja, o que o cliente diz que quer; 2) as reais, ou o que o cliente realmente quer; 3) as não declaradas, ou o que o cliente espera receber; 4) as de algo a mais, ou o que o cliente gostaria; 5) as secretas, ou de como o cliente quer ser visto pela sociedade.

Mas para identificar as necessidades, devemos, conforme Bortoli (2022), seguir alguns pontos, como: 1) fazer pesquisas com frequência, criando uma rotina de pesquisas com o público-alvo; 2) estarmos atentos aos *feedbacks* que os clientes trazem; 3) nos colocar no lugar do cliente; e, também, 4) interagir com o público-alvo nos canais de comunicação disponíveis.

Com vistas nestes pontos teóricos, analisando as respostas dos entrevistados, percebemos que suas necessidades são:

## Requisitos Funcionais

Os requisitos funcionais descrevem as funcionalidades que cada nível de usuário deve ser capaz de acessar no sistema (SOMMERVILLE, 2011). Esses requisitos estão relacionados com as funções que o aplicativo realiza.

* Credenciais do usuário:

Criação, edição e exclusão de usuário na plataforma, de acordo com a função que será desempenhada por "promotor" ou "supervisor", coletando as seguintes informações "Nome", "Usuário", "Foto" e solicitando a definição de uma senha segura.

* Validar credenciais:

Checar a situação do usuário inserindo se "ativo" ou "inativo" e realizar a validação da senha inserida, apresentando os módulos de acordo com a função desempenhada.

* Cadastrar bandeiras:

Realizar o cadastro das bandeiras inserindo uma "Descrição" simples, função liberada somente ao usuário *'back office".*

* Cadastrar lojas:

Realizar o cadastro das lojas que serão visitadas pelos promotores e supervisores, inserindo a "Razão Social", "Nome Fantasia", "CNPJ", "Endereço" e "CEP", função liberada somente ao usuário *"back office".*

* Vincular lojas:

Vincular a loja, no momento do cadastro, a uma bandeira previamente cadastrada utilizando o "Código da Bandeira", função liberada somente ao usuário *"back office".*

* Cadastrar marca:

Realizar o cadastro das marcas dos produtos, inserindo uma "Descrição" simples, informar se "ativo" ou "inativo".

* Coletar dados:

Inserir dados dos produtos, como preço, prateleira de exposição e informações adicionais caso necessário.

## Requisitos não-Funcionais

Os requisitos não-funcionais não estão diretamente conectados com as funções do sistema; no entanto, são restrições impostas aos serviços oferecidos pelo sistema (SOMMERVILLE, 2011).

* Acessos simultâneos:

Acesso de 100 pessoas simultaneamente, por limitação do banco de dados, utilizado a versão *free.* Caso haja necessidade de escalar a quantidade de acessos simultâneos, será preciso contratar plano pago.

* Compatibilidade:

O aplicativo deve ser acessado por um aparelho móvel com o sistema operacional Android, caso acessado por outros sistemas operacionais poderão ocorrer incompatibilidade de funcionalidades.

# DESENHO DO SISTEMA

O desenho do sistema é composto pela prototipação, diagramas, personagens, tabelas, entre outras ferramentas que mostram como será a composição do *software* e as hierarquias que ele corresponderá.

Mattar (2010) sugere pensar no desenvolvimento de um aplicativo seguindo três principais: pré-produção, produção e pós-produção. Na pré-produção, é desenvolvido o conceito e o design do aplicativo; na produção, ocorre sua implementação; e na pós-produção, os testes.

* 1. pERSONAGENS

Os personagens ou atores são pessoas que utilizarão o aplicativo com determinadas responsabilidades e acessos ao sistema, os atores terão acesso ao sistema:

Analista de Tecnologia: O analista é o responsável pelas correções de problemas, melhorias do aplicativo e prestará o suporte ao “supervisor” e o “promotor” em questões de incompatibilidade e erros.

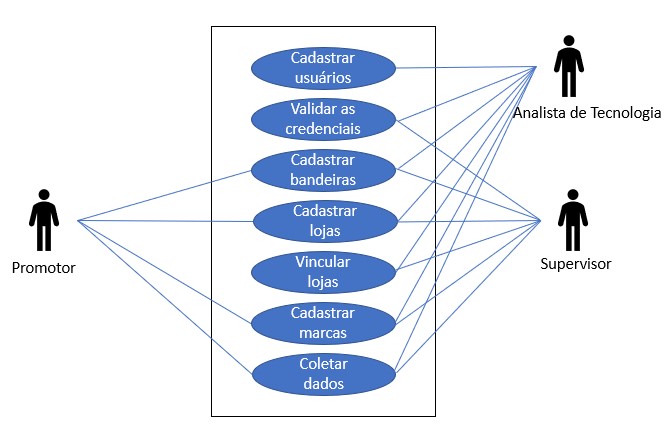
Supervisor: O supervisor será responsável por validar e vincular as lojas e bandeiras para que o promotor consiga realizar a coleta das informações nos estabelecimentos.

Promotor: Será responsável por cadastrar a loja, bandeira e marca para realizar a coleta de dados nos estabelecimentos.

* 1. diagrama de caso de uso

Como foram definidos os requisitos funcionais e os atores, é possível modelar as funcionalidades do aplicativo proposto. Dos três atores do sistema (promotor, supervisor e analista de tecnologia), apenas o analista de tecnologia tem acesso a todas as funcionalidades do sistema. O supervisor somente não tem o acesso de cadastro de usuários que fica sob responsabilidade do analista de tecnologia.

**Figura 3** - Diagrama de caso de uso.



Fonte: De autoria própria.

## Dicionário de dados

Tabelas que expressam as tabelas existentes no banco de dados. Contendo os tipos dos campos, tamanho dos campos e a relação da tabela com as demais; assim como as restrições de cada campo existente.

**Tabela 1** – Tabela de pessoa.



**Tabela 2** – Tabela de perfil.



**Tabela 3** – Tabela de bandeira.



**Tabela 4** - Tabela de loja.



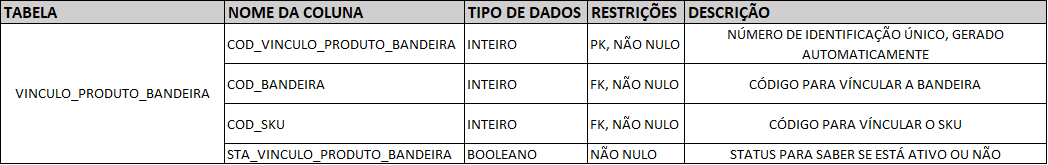
**Tabela 5** - Tabela de sku.



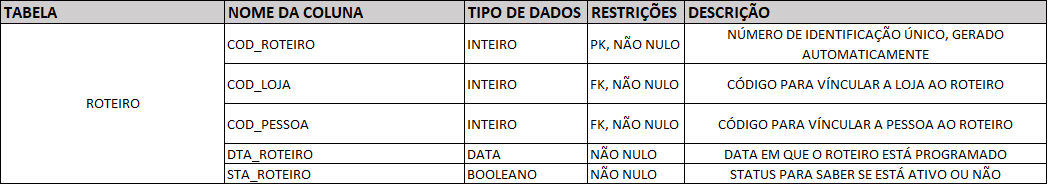
**Tabela 6** – Tabela de marca.



**Tabela 7** – Tabela de vínculo produto bandeira.



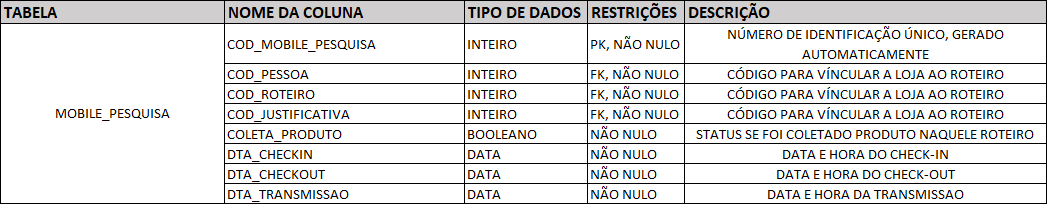
**Tabela 8** – Tabela de roteiro.



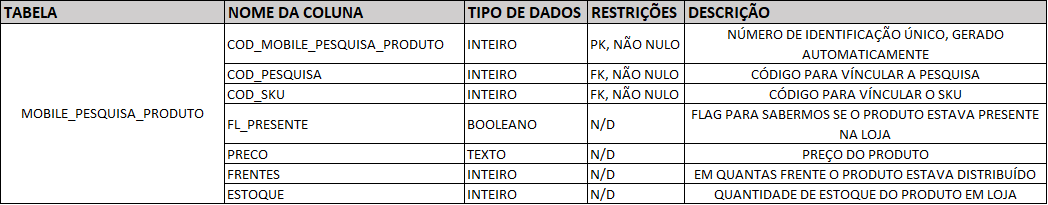
**Tabela 9** – Tabela de justificativa.



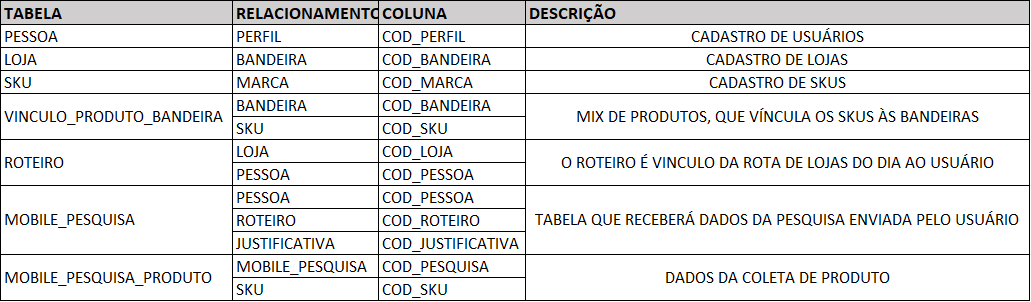
**Tabela 10** – Tabela de mobile pesquisa.



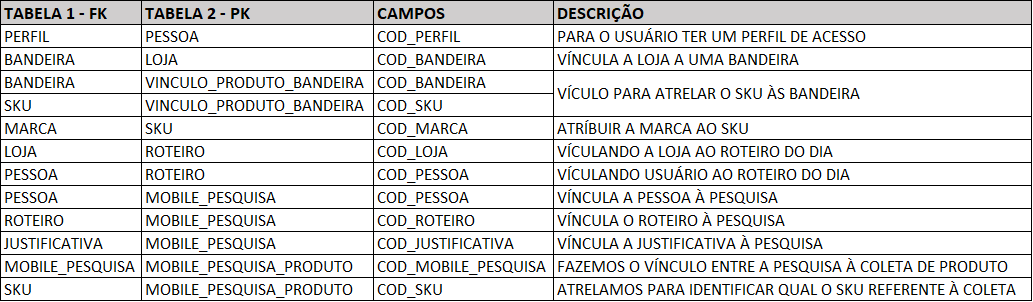
**Tabela 11** – Tabela de mobile pesquisa produto.



**Tabela 12** – Tabela de relacionamentos.



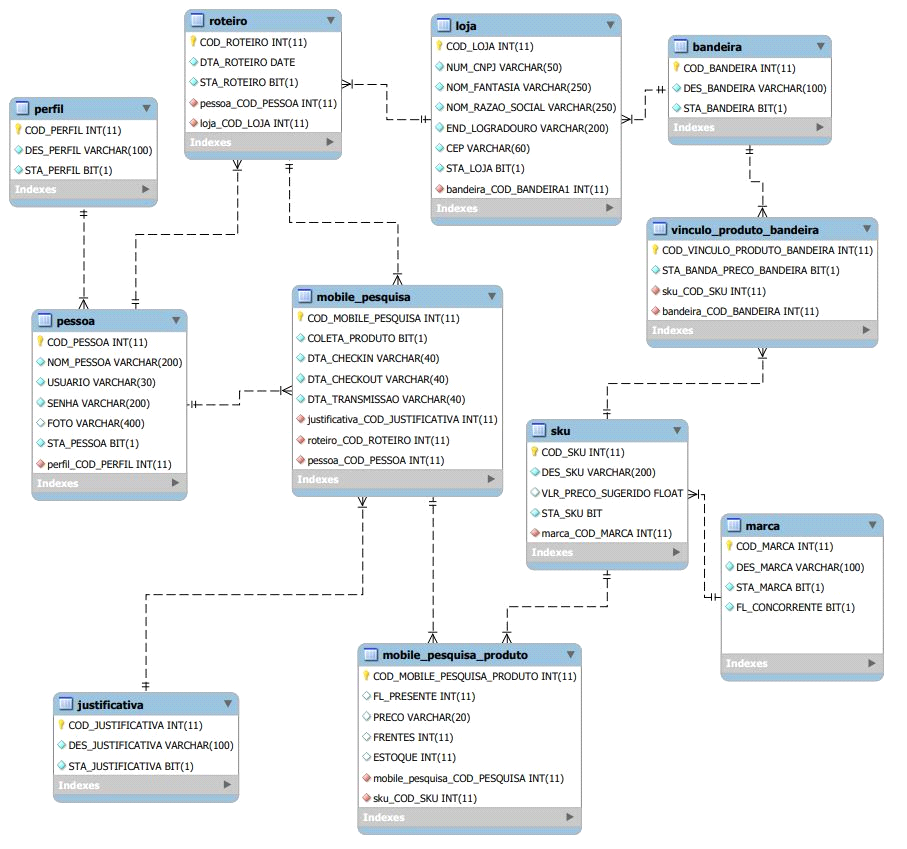
**Tabela 13** – Tabela de detalhamento de relacionamentos.



* 1. Modelo de entidade lógica

O modelo lógico tem como objetivo representar as estruturas que irão armazenar os dados dentro de um Banco de Dados, a partir deste momento é que são definidas com maior propriedade as entidades e os seus atributos. Na Figura 4 abaixo, temos o modelo lógico desenvolvido para o nosso aplicativo.

**Figura 4** - Modelo lógico.



Fonte: De autoria própria.

# ARQUITETURA DO SISTEMA

A arquitetura do sistema é por qual padrão o sistema/aplicativo foi desenvolvido, seguindo princípios e diretrizes, onde não se aprofunda em detalhes tecnológicos, mas atender o que realmente o cliente precisa, oferecendo uma visão genérica do sistema a ser criado.

O padrão de arquitetura utilizado para construir o Kotler foi o padrão MVC *(Model-View-Controller).* O MVC é uma sigla do termo em inglês *Mode*l (modelo) *View* (visão) e *Controller* (Controle) que facilita a troca de informações entre a interface do usuário aos dados no banco, fazendo com que as respostas sejam mais rápidas e dinâmicas. Além disso, esse padrão proporciona estabilidade no processo de comunicação entre seus elementos de maneira dinâmica para que a experiência do usuário não seja prejudicada.

## Projeto de interface do sistema

A interface foi idealizada para ter uma utilização fácil e prática para os nossos usuários, que serão principalmente os promotores e secundariamente os seus supervisores. A fim de deixar a coleta dos dados o mais eficiente possível, iremos ter um menu enxuto, assim como uma tela de coleta com nomes claros para serem de fácil entendimento.

Para construção das telas, utilizamos o padrão recomendado pela google, fazendo uso do *view binding*, que é um recurso que facilita a programação de códigos que interagem com visualizações, que por sua vez compõem as telas ou como são mais conhecidas as *Activities.*

Optamos pela utilização da *bottom navigation* no sistema de menu, exatamente por enquadrar diferentes menus em um espaço único, sempre visível e de fácil acesso ao usuário; assim facilitando a utilização, pois muitos já vão estar familiarizados, já que hoje grandes aplicativos a utilizam como, por exemplo, o próprio Instagram.

Teremos inicialmente, após o login, três telas com funcionalidade simples e intuitivas, e são elas: tela de sincronização, tela de perfil, e a tela de roteiro. A tela de sincronização é a primeira tela a ser visualizada pelo usuário após o login, isso porque a sincronização é imprescindível para o funcionamento do app, já que conseguimos trabalhar com ele offline, com a necessidade de que o usuário realize a sincronização dos dados.

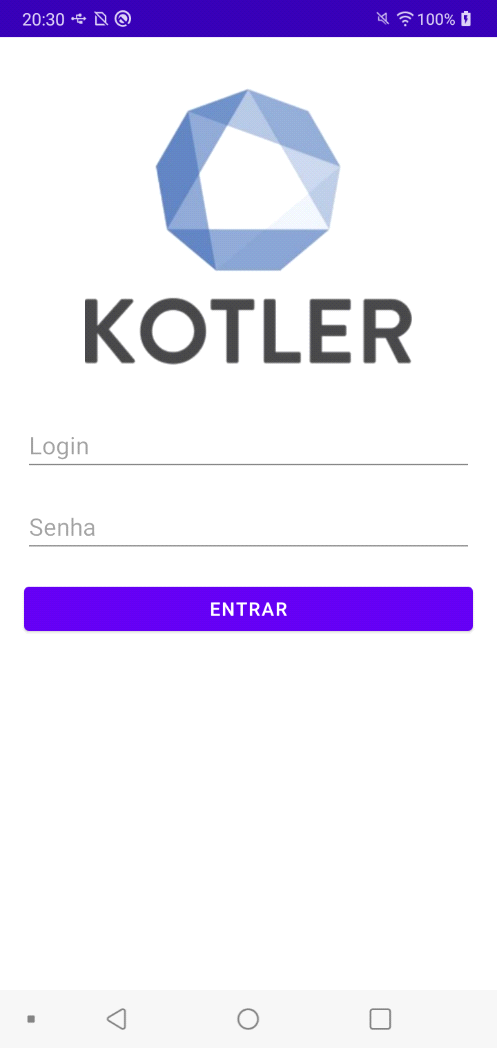
A segunda tela que temos é a tela de perfil, onde usuário poderá dispor de informações básicas e poderá deslogar do sistema caso necessário. E a terceira tela que temos na seção inicial do app é a tela de Roteiro, onde se encontra a lista de lojas diária do usuário.

Ao realizar check-in em uma das lojas, teremos a lista de módulos de coleta e os botões de justificativa, *check-out* e transmissão, para finalização da pesquisa.

### Layouts de telas e descrição

A primeira tela com qual o usuário irá se deparar, é a tela de login, onde ele deverá estar previamente cadastrado pelo Back Office da operação, pois não é oferecido opção de cadastro de novos usuários no app.

**Figura 5** – Tela de login.

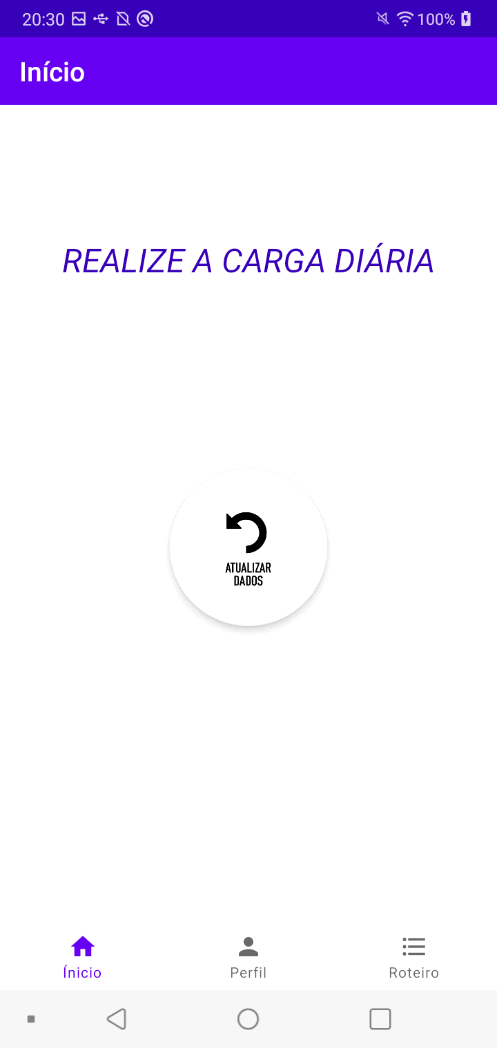


Fonte: Aplicativo Kotler.

Assim que realizado o login no app, o usuário irá visualizar a tela inicial, contando com o menu que fica na parte inferior da tela. Nesta tela inicial, ele conseguirá navegar pelas seguintes telas: início, perfil e roteiro.

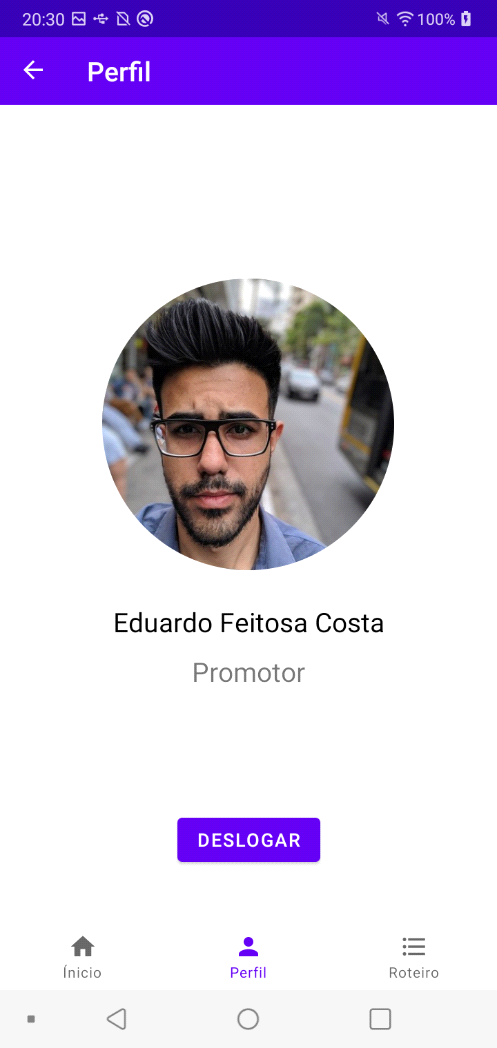
Na tela de início ele poderá realizar a sincronização dos dados, ou seja, receber o roteiro de lojas do dia, com as coletas que irá ter de realizar.

**Figura 6** - Tela de início.



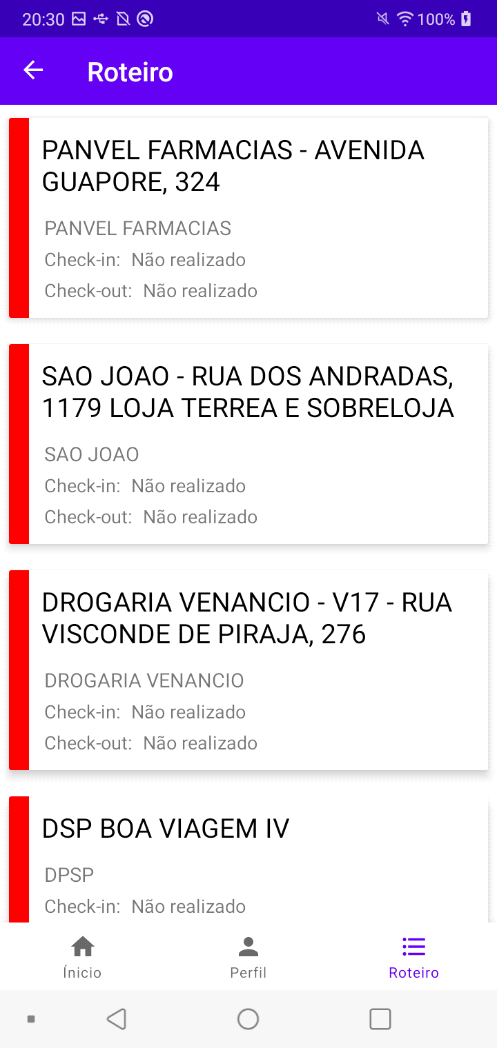
Fonte: Aplicativo Kotler.

**Figura 7** - Tela de perfil.



Fonte: Aplicativo Kotler.

**Figura 8** - Tela de roteiro.



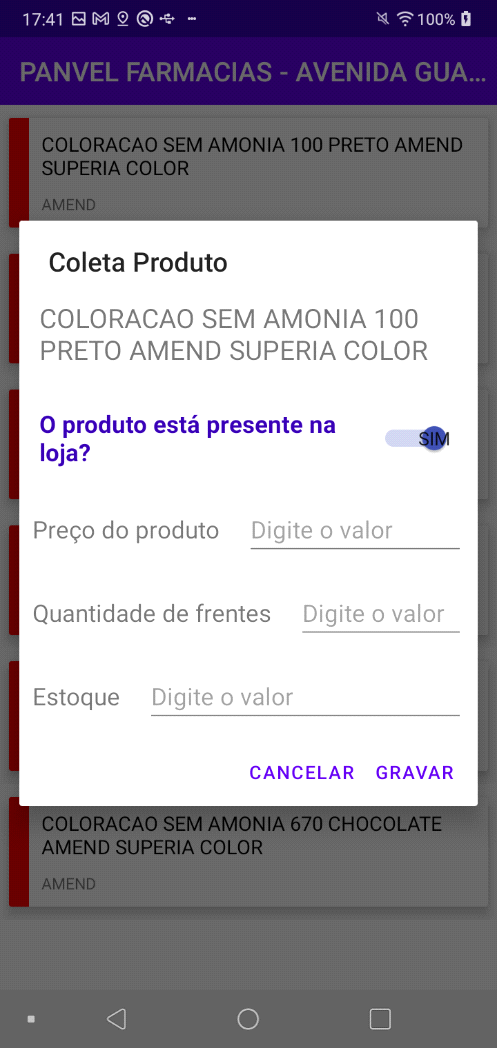
Fonte: Aplicativo Kotler.

Ao realizar o check-in, o promotor iniciará sua pesquisa, ele verá os módulos a serem coletados naquelas lojas, e no menu inferior poderá visualizar três botões para controle de estadia em loja e pesquisa.

No módulo de produto temos uma lista com os produtos a serem coletados na loja, ao iniciar a coleta de alguns deles, irá abrir uma caixa de diálogo com a informações a serem coletadas.

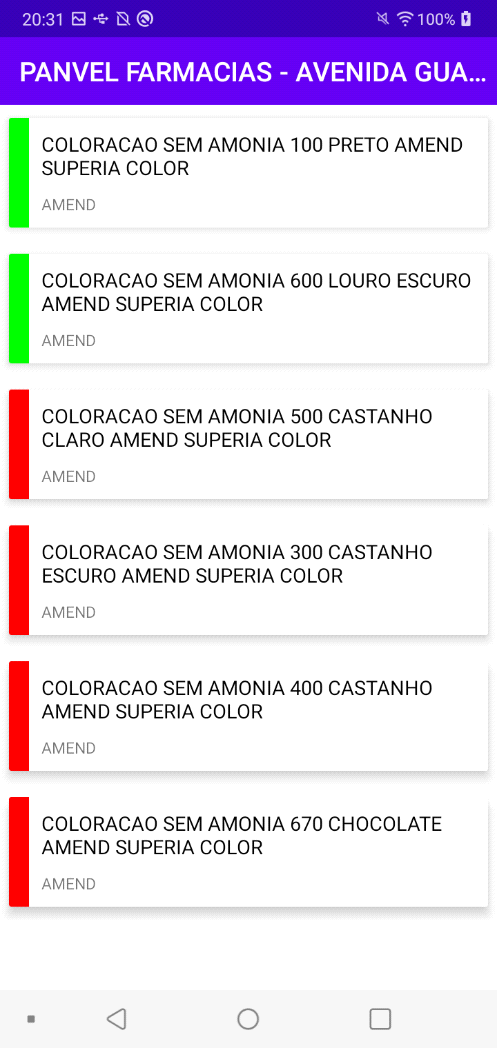
Temos indicadores na lista de roteiro, onde vermelho são para lojas onde não foram realizadas o check-in como as lojas da Figura 8, amarelo em lojas onde já foi realizado check-in, porém ainda não tiveram suas informações transmitidas e verde quando as informações da pesquisa já foram transmitidas como mostra na Figura 13.

**Figura 9** - Tela de coleta.



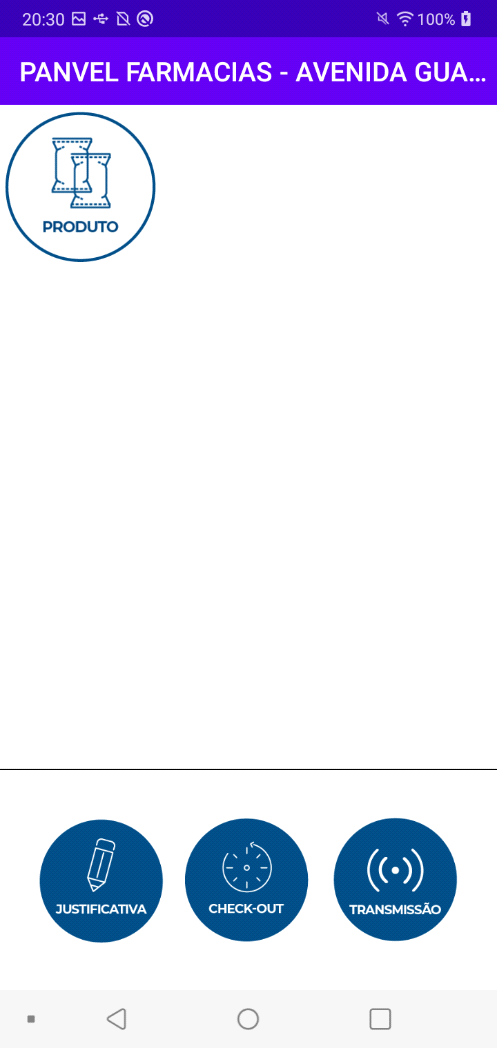
Fonte: Aplicativo Kotler.

**Figura 10** - Coleta de produto.



Fonte: Aplicativo Kotler.

**Figura 11** - Diálogo produto.



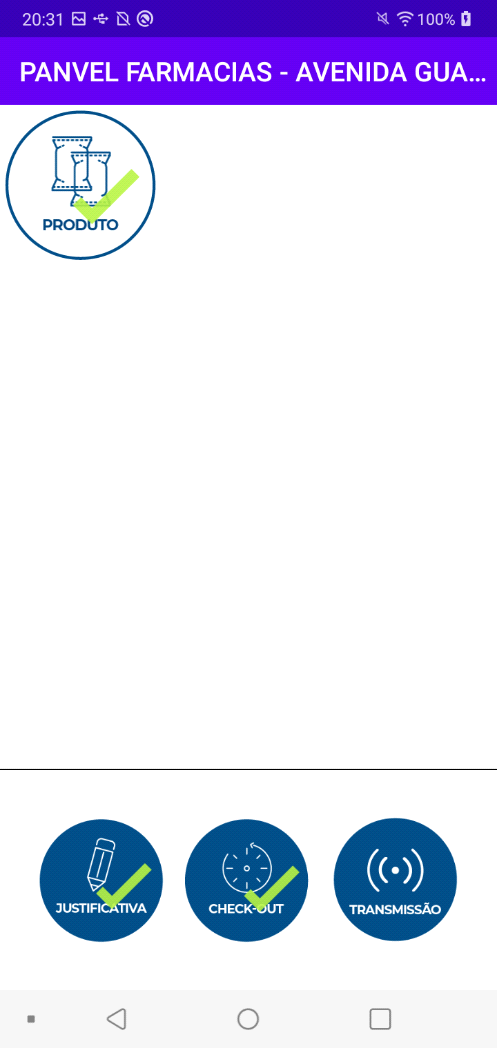
Fonte: Aplicativo Kotler.

Uma premissa semelhante quando o design foi levado em consideração na coleta de produto (Figura 10), onde os indicadores funcionam de forma parecida com os de roteiro, porém somente com as opções coletado e não coletado, ou seja, verde e vermelho respectivamente.

Quanto aos botões implementamos indicadores, para que o usuário tenha um respaldo de em qual estágio está a sua pesquisa, ao justificar a loja, ou ao realizar *check-out*, ele saberá que o realizou, pois, a interface irá sinalizar sobre o botão, assim como quando ele tiver finalizado de preencher todos os itens da coleta., como podemos ver na Figura 12.

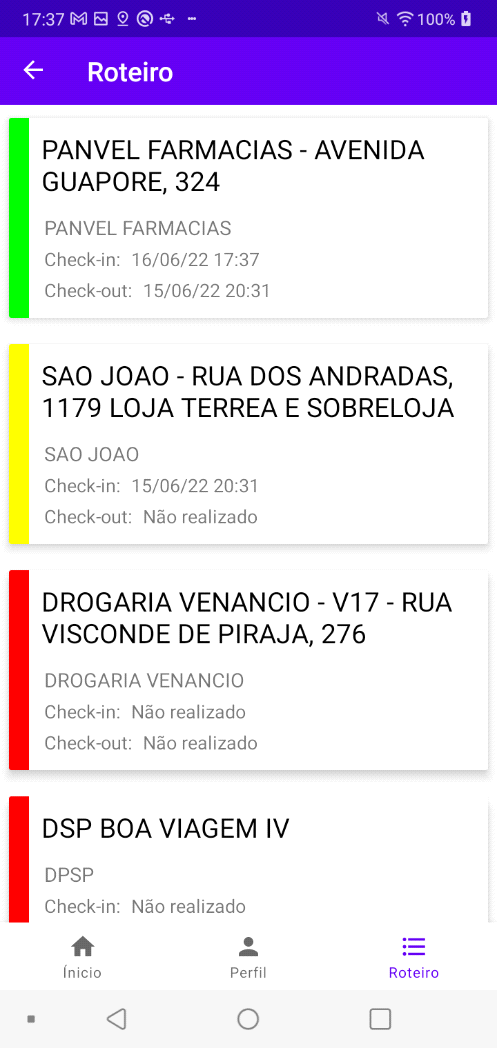
Optamos por deixar o visual simplificado e intuitivo quando aos estágios de pesquisa, pois assim iremos fornecer um app de fácil utilização e com facilidade na hora de implementar em uma operação.

**Figura 12** - Tela de coleta com indicadores.



Fonte: Aplicativo Kotler.

**Figura 13** - Tela de roteiro com indicadores.



Fonte: Aplicativo Kotler.

# 7CONCLUSÃO (Numerar ítem 7) Refazer a Conclusão....o TCC não apresenta um desfecho conclusivo) Utilizem uma página com aprox.3 paragrafos para desenvolver a conclusão.

Concluímos através deste trabalho que......

De acordo com as funcionalidades e a tecnologia utilizada entendemos que....

A tecnologia de XYZ..... permite.....

As funcionalidades desta aplicação permitem.......

A melhor linguagem para se cumprir esta funcionalidade foi ou foram......

Quais problemas esta ferramenta resolve?

Este trabalho foi desenvolvido com o intuito de facilitar a relação de coleta de dados dos produtos de merchandising pelos promotores, que no passado recente costumava realizar a coleta dos dados através de pranchetas, no entanto, a utilização do aplicativo oferecerá facilidade e praticidade na utilização dos dados coletados e diminuirá as informações pedidas.

As ideias que foram pensadas, e não incluídas no aplicativo atual, mas podem ser implementadas futuramente são: adaptação para outros sistemas operacionais; fotos da prateleira onde o produto está localizado; leitura do código de barra do produto através da câmera.

Concluímos através deste trabalho que cada vez mais a tecnologia está definindo o rumo das organizações e as operações logísticas está sendo crucial para diminuir os custos e aumentar a competividade no mercado. A construção desse aplicativo foi para que os promotores de produtos pudessem mais ágeis e

# REFERÊNCIAS BIBLIOGRÁFICAS

CARVALHO, Marcos Roberto. **Gestão dos Canais de Distribuição.** – Curitiba:IESDE Brasil S.A., 2009.

GOODRICH, Michael T; TAMASSIA, Roberto. **Estruturas de dados e algoritmos em Java**. 4.ed. Porto Alegre: Bookman, 2007.

FERNANDES, Kleber dos Santos. **LOGISTICA: Fundamentos e processos.** – 1 ed. rev.. – Curitiba, PR:IESDE Brasil, 2012.

MACHADO, Jacira de Lima. **O papel do serviço público municipal na logística reversa do óleo de cozinha.** – São Paulo, SP: [s.n], 2009.

MENDES, Eduardo Fernando. **Automatização da Técnica de Mineração de dados auxiliada por guias.** 2009. 101 f. Dissertação (Mestrado) - Curso de Ciência da Computação, Metodista de Piracicaba, Piracicaba, 2009.

REZENDE, Solange Oliveira**. Mineração de Dados. In: Congresso da sociedade Brasileira de Computação**, 25., 2005, São Leopoldo, Rs. A Universidade da Computação: Um agente de inovação e Conhecimento. São Leopoldo, Rs: Unisinos, 2005. p. 397 - 433.

ROSA, Rodrigo de Alvarenga. **Gestão de operações e logística I.** Florianópolis:Departamento de Ciências da Administração / UFSC; [Brasília]: CAPES: UAB, 2011.

SANTOS, Daiana Pereira dos. **Uso de Data Mining em Base de Dados de Empresa Calçadista.** 2007. 49 f. Trabalho de Conclusão de Curso (Graduação) - Curso de Ciências da Computação, Centro Universitário Feevale, Novo Hamburgo, Rs, 2007.

SOARES, Bruno Augusto Lobo. **Aprendendo a linguagem PHP.** São Paulo: Ciência Moderna, 2007.

SOMMERVILLE, I**. Engenharia de Software-9a Edição.** [S.l.]: Pearson Education, 2011.

TABOADA, Carlos**. Gestão de Tecnologia e Inovação na Logística.** – Curitiba: IESDE Brasil S.A.,2009.

DEVELOPER, Android, 2022**. Documentação do Kotlin**. Disponível em: <https://developer.android.com/kotlin?hl=pt&gclid=cj0kcqjwuo6wbhdlarisaideydkrxgrdzprbywi7ztj3hwqguy7pttera\_lna9txbrgc1l-khwje7jcaauctealw\_wcb&gclsrc=aw.ds>. Acesso em: 05 maio. 2022.

MORAIS, E. A. M.; AMBRÓSIO, A. P. L. **Ontologias: conceitos, usos, tipos, metodologias, ferramentas e linguagens.** Relatório Técnico, Dezembro, 2007. Disponível em < http://www.inf.ufg.br/sites/default/files/uploads/relatorios-tecnicos/RT-INF\_001- 07.pdf>. Acesso em 25 Abril. 2022.