

FICOS

APLICAÇÃO MOBILE PARA ANÚNCIO DE PRODUTOS

Maicon Douglas Filipiaki

Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil
mdfilipiaki@gmail.com

Flávia Pereira de Carvalho

Professora Orientadora

Faculdades Integradas de Taquara – Faccat – Taquara – RS – Brasil
fpereira@faccat.br

Resumo

Este presente artigo disserta sobre a elaboração do aplicativo FiCos. Com a atual crise sanitária devido a pandemia do Coronavírus, muitas pessoas precisam ficar em suas casas, aumentando o número de compras pela internet. Pensando nisso, foi desenvolvido o FiCos, aplicativo que tem como objetivo auxiliar o comércio local a vender para as pessoas que estão dentro do seu raio de ação. O desenvolvimento do aplicativo foi feito utilizando Flutter, um framework desenvolvido pela Google para criação de aplicativos.

Palavras-chave: Anúncio. Aplicativo Móvel. Comércio local. Venda.

MOBILE APPLICATION FOR PRODUCTS ADVERTISING

Abstract

This article discusses the development of the FiCos application. With the current health crisis due to the Coronavirus pandemic, many people need to stay in their homes, increasing the number of purchases over the internet. With that in mind, FiCos was developed, an application that aims to help local businesses to sell to people who are within their scope of action. The application was developed using Flutter, a framework developed by Google for creating applications.

Keywords: *Announcement. Mobile Application. Local trade. Sale..*

1 INTRODUÇÃO

Uma das principais atividades que aumentaram com a atual crise sanitária do covid-19, principalmente nos grandes centros urbanos, foi a venda de produtos pela internet, produtos de variados tipos, desde comida a coisas mais comuns. Cerca de 7,3 milhões de brasileiros fizeram compras, ou sua primeira compra pela internet, o que veio a ocasionar em um crescimento de 40% no setor de vendas on-line. Através disso o Brasil conseguiu alcançar a marca de 41 milhões de usuários ativos no e-commerce (INGIZZA, 2020).

Uma grande parcela dos estabelecimentos locais está lutando contra esse comércio on-line, ou seja, estão tentando vender seus produtos sem contar com o suporte de um site ou de um aplicativo para que as pessoas possam conhecer seus produtos. Normalmente empresas locais e menores acabam utilizando uma forma de marketing mais convencional, com panfletos, carros de som, banner etc, o que acaba sendo anulado pelo momento da pandemia, onde pessoas param de circular nas vias, fazendo com que as formas de marketing utilizadas por elas, tenham seus efeitos diminuídos.

A relação entre marketing digital e quantidade de venda na pandemia está conectada diretamente. A transformação digital está sendo falada desde os anos 2000, da bolha das startups, e chegamos em um momento da história humana, onde empresas que não tiverem como anunciar seus produtos de forma digital vão perceber seus rendimentos e vendas tendo uma queda. A transformação digital vem para aumentar ainda mais o alcance de empresas menores, das quais são responsáveis por 72% dos empregos do país no atual momento, normalmente empresas locais ficavam limitadas dentro de seus bairros ou alguns quilômetros, principalmente porque era até onde seu marketing chegava (ROCHA, 2020).

Com a aceitação de empresas que precisam entrar de cabeça no meio digital, algumas perceberam a falta de um aplicativo onde consigam fazer seus anúncios e também possam sanar dúvidas de seus possíveis clientes. Algumas empresas acabam não tendo dinheiro para construir uma plataforma da qual consigam gerir isso, o que acaba gerando uma lacuna entre empresa e o meio digital, da qual normalmente é utilizado o nome de marketplace, que serve como um catálogo on-line, onde constam itens de várias empresas.

2 JUSTIFICATIVA

Em 2019 cerca de 41% dos trabalhadores estão no mercado de trabalho de forma autônoma sem nenhuma forma de auxílio governamental (RIVEIRA, 2020). Em conversas com amigos, vários relatam problemas para localizar determinados estabelecimentos e como alguns são donos de algumas lojas relataram o oposto, onde não conseguiam ter uma rotatividade muito grande de clientes. Para tentar sanar o problema de ambas partes surgiu a ideia de criar e desenvolver um aplicativo com a capacidade de aproximar ambos, trazendo a possibilidade de clientes encontrarem novos estabelecimentos e mercadorias e também proporcionar aos estabelecimentos terem uma rotatividade maior de clientes.

3 REFERENCIAL TEÓRICO

3.1 Visão geral sobre marketplace

O marketplace apareceu no Brasil em torno dos anos de 2012, o qual ficou conhecido por ser um shopping center na web, onde é possível, normalmente, encontrar vários tipos de produtos. É considerado vantajoso tanto para o cliente quanto para o anunciante, pois o cliente encontra diversos produtos e marcas em um único lugar, e o anunciante consegue ter seu negócio na internet de uma forma mais simples e barata (ZACHO, 2017). Um e-commerce passa a ser considerado um marketplace após começar a operar tanto produtos próprios e produtos provenientes de terceiros. (SAMPAIO, 2018) A partir desse ponto pode-se entender que o aplicativo FiCos se enquadra como um marketplace, pois por mais que o mesmo não ofereça produtos próprios, oferece produtos de terceiros.

Uma pesquisa realizada pela Ebit/Nielson mostra que no mundo inteiro os marketplaces vêm crescendo cada vez mais. Dois países se destacam muito pelo alto crescimento que são Brasil e Espanha. No Brasil, pegando somente o ano de 2019, os marketplaces tiveram um crescimento de 13% se comparado com os 12% de e-commerce normais. O faturamento por sua vez foi de R\$17,6 bilhões com vendas de um ticket médio de R\$489,00 (RONDINELLI, 2019).

Os comerciantes se voltaram para marketplace normalmente por causa de uma queda de vendas ou somente para aumentar a quantidade de vendas de seus estabelecimentos, mas os marketplaces contêm mais vantagens do que somente um site de venda, das quais são (SPONCHIADO, 2015):

1. Visibilidade: Normalmente ao cadastrar produtos em um marketplace, seu produto automaticamente ganha mais visibilidade pois acaba atingindo públicos diferentes e possivelmente de outros lugares.
2. Baixo Investimento: Normalmente os valores que os marketplaces cobram são em cima das vendas, uma margem da qual entra normalmente a tarifa de transições de cartão e uma parte que seria de custos. Mas o FiCos, aplicativo desenvolvido neste trabalho, não terá custos para os comerciantes.
3. Alto retorno: Normalmente o custo dos marketplaces é por fazer toda a questão de manter e atualizar o site onde são anunciados os produtos. Então o custo que é repassado para o comerciante acaba sendo bem mais baixo do que ter um e-commerce.
4. Relevância SEO: Ao anunciar em um marketplace seu estabelecimento automaticamente acaba gerando um SEO(Search Engine Optimization, otimização para mecanismos de busca) muito melhor dentro das buscas de buscadores como Google, Bing e Yahoo, pois normalmente os marketplaces são desenhados e desenvolvidos para ter o maior engajamento possível nos buscadores atuais.

4 METODOLOGIA

Para manter um desenvolvimento planejado e organizado foram utilizados os conceitos e técnicas indicados pela Metodologia Ágil.

Metodologia Ágil são formas e técnicas para gestão de um projeto da qual oferece mais rapidez, eficiência e flexibilidade. Foram inicialmente feitas para a área de tecnologia mas hoje já ultrapassaram essa barreira e estão sendo utilizadas em quase todos os setores (PEREIRA, 2020).

A utilização da metodologia ágil está sendo empregada em projetos normalmente pelos seguinte fatores: (PEREIRA, 2020)

- Desenvolvimento incremental, da qual sempre buscando coisas novas e melhorias
- Entregas menores e rápidas.
- Capacidade de tornar o escopo do projeto mais flexível.
- Agregação de valor conforme a necessidade do cliente.
- Adaptação às mudanças de escopos e de tarefas rápidas.

Dentre as variedades de metodologias ágeis, como o Kanban, Scrum, Dynamic System Development Model (DSDM), Extreme Programming (XP) e Feature Driven Development (FDD), optou-se pela utilização do método Kanban. Para esse projeto foi escolhido essa

metodologia por vários motivos alguns deles são um pré conhecimento, e principalmente por ser uma metodologia da qual ao olharmos pro board onde contém nossos afazeres conseguimos ter uma noção prévia se o andamento do projeto está adiantado, atrasado ou está sendo elaborado conforme programado.

A palavra Kanban vem da cultura asiática, mais precisamente do Japão, onde Kan se refere a visual e Ban seria cartao, entao formando a palavra “cartões visuais”. Os pontos mais vantajosos pelo uso do KanBan e porque ela é muito simples de usar, atualizar e de se entender, pois por serem cartões distribuídos dentro de colunas. (PAREDE, 2019)

4.1 Análise

A análise do projeto, que resultou na construção do aplicativo FiCos, foi dividida em etapas.

Na primeira, foi levantado o problema, que consiste na necessidade de auxiliar comerciantes locais que precisam de uma forma de anunciar seus produtos e assim atingir um público maior e aumentar suas vendas.

Na segunda etapa, foram enumerados os requisitos funcionais e não funcionais, necessários para a resolução da aplicação. Com esses requisitos foram elaborados os casos de usos. Ainda nesta fase, foram realizados o diagrama de Caso de Uso e os diagramas de Atividades, referentes aos Requisitos Funcionais (RF) dos quais consistem em:

- RF01 - Login: Usuário deve ser capaz de acessar a aplicação, e ter acessos a suas mensagens, suas pronta entregas e seus itens;
- RF02 - Cadastro: Usuário deve ser capaz de se cadastrar no sistema utilizando tanto sua rede social do Facebook quanto por seus dados;
- RF03 - Cadastro de Pronta Entrega e Itens: Usuário que estiver no papel de vendedor deve ser capaz de cadastrar suas pronta entregas e dentro de suas pronta entregas poder cadastrar itens;
- RF04 - Edição de Pronta Entrega e Itens: Usuário no papel de vendedor deve ser capaz de editar e deletar suas pronta entregas e seus itens;
- RF05 - Listagem: Usuários independentemente do seu papel, tanto de cliente quanto vendedor, deve ser capaz de ver itens na listagem do catálogo, dos quais podem ser entregues na sua área.

- RF06 - Chat: Usuários que estiverem exercendo o papel de clientes e buscando itens, devem ser capazes de visualizar um produto e entrar em contato com o vendedor para obter mais detalhes do produto.
- RF07 - Opinar sobre cliente: Usuários que estiverem executando o papel de vendedores devem ser capazes de avaliar clientes;
- RF08 - Opinar sobre vendedores: Usuários que estiverem executando o papel de clientes devem ser capazes de avaliar vendedores;

A terceira fase, foi destinada ao desenvolvimento do software. Para cada RF levantado na etapa anterior, obtiveram suas funcionalidades e particularidades desenvolvidas e após o término de alguma tarefa a mesma era testada para ver se o comportamento era o esperado.

Na última etapa do projeto, a fase final do projeto, coube testar usabilidades e busca de melhorias de performance e correção de alguns bugs que se apresentaram ao testar a aplicação como um todo.

4.2 Modelagem

A modelagem de software é um forma de simplificar a realidade da qual vivemos, para tentarmos representar na forma de software mais próxima possível. A partir dos modelos e mapas que construídos, é possível ter uma clara visão das funcionalidades do software em questão e como deve ser desenvolvido. Também é possível especificar estruturas ou até mesmo comportamentos esperados pelo sistema e, através disso conseguir desenvolver de um forma orientada ao problema que o sistema se propõe em resolver e tomar decisões de uma forma mais lúcida (CALSAVARA, 2019).

No atual projeto foi utilizado um dos padrões mais conhecido e utilizado no desenvolvimento de software que é o Unified Modeling Language, conhecido pela sigla UML. Conforme Guedes informou no seu livro¹ “É uma linguagem visual utilizada para modelar software baseados no paradigma de orientação a objetos. É uma linguagem de modelagem de propósito geral que pode ser aplicada a todos os domínios de aplicação”

¹ Livro UML 2 Uma Abordagem Prática.

Para a modelagem dos diagramas foi utilizada a ferramenta Lucid. O primeiro diagrama a ser desenvolvido foi o de Diagrama de Caso de Uso, pois de acordo com Guedes (2011), diagrama de casos de uso é o diagrama mais essencial para início do desenvolvimento de um software, pois ele apresenta de uma forma simples e de fácil compreensão os comportamentos e funcionalidades que o software deve conter.

Figura 1, Diagrama de Caso de Uso (Use Case - UC), para cada UC representa um requisito funcional do aplicativo desenvolvido neste trabalho.

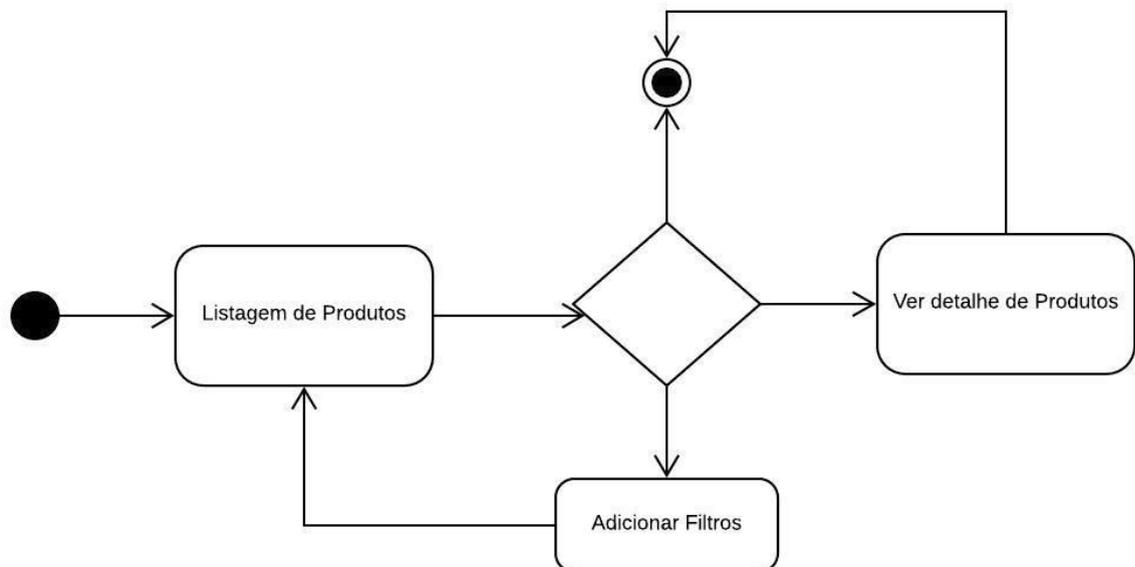
Figura 1 – Diagrama Caso de Uso.

Fonte: Autor.



Figura 2, exibe o diagrama de atividade do caso de uso de Consultar Itens. Nesse diagrama é apresentada a sequência do funcionamento da consulta de itens. Na página inicial do aplicativo, serão apresentados alguns itens próximos a localização do usuário. O usuário pode selecionar alguns filtros para deixar assim sua consulta mais específica para o que o mesmo precisa no momento.

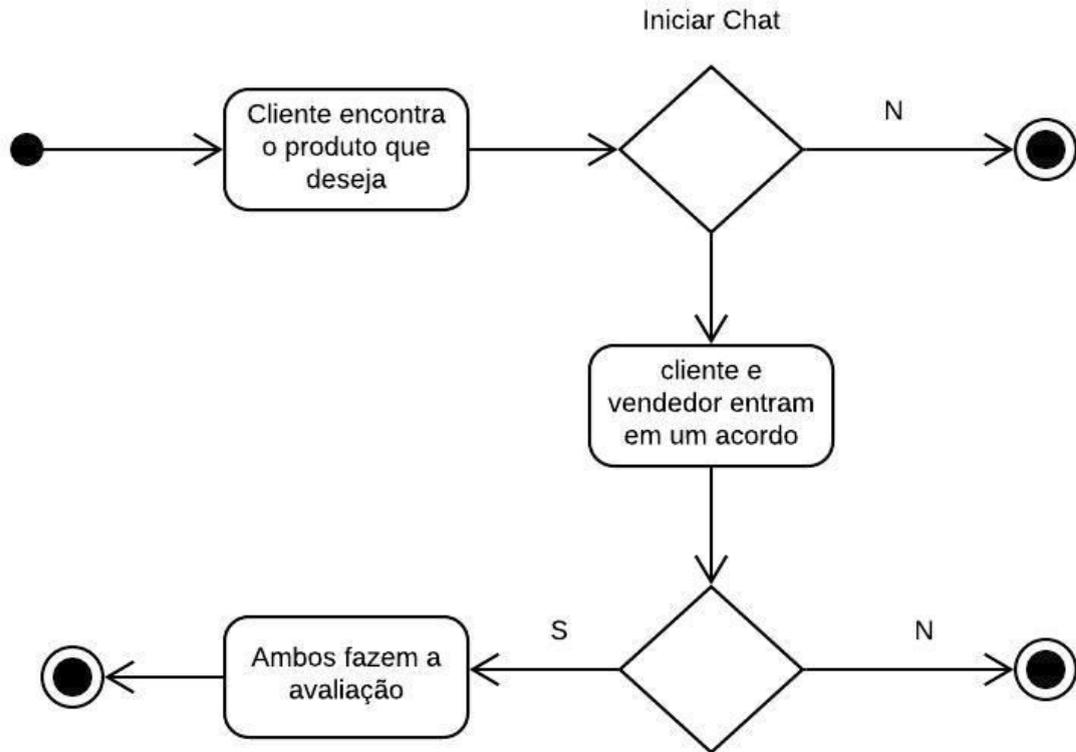
Figura 2 – Diagrama de atividade -Consultar itens.



Fonte: Autor.

Figura 3, diagrama de atividade do requisito Funcional de Conversa entre cliente e vendedor. O cliente busca o produto que deseja encontrar, onde terá um botão para iniciar uma conversa com o vendedor, fazendo o seu pedido ou até mesmo podendo negociar com o vendedor o valor e, após isso, ambos podem se avaliar.

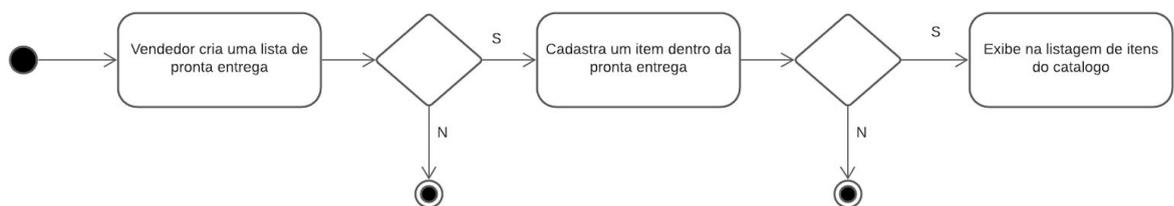
Figura 3 – Diagrama de Atividade – Conversa entre cliente e vendedor.



Fonte: Autor.

Figura 4, diagrama de atividade do UC, Cadastro de um item. Para usuários que são do tipo vendedor, o mesmo pode criar listas de pronta entregas onde estarão contidos os itens. O vendedor deve obrigatoriamente cadastrar os itens dentro das pronta entregas.

Figura 4 – Diagrama de Atividade – Cadastro de item.



Fonte: Autor.

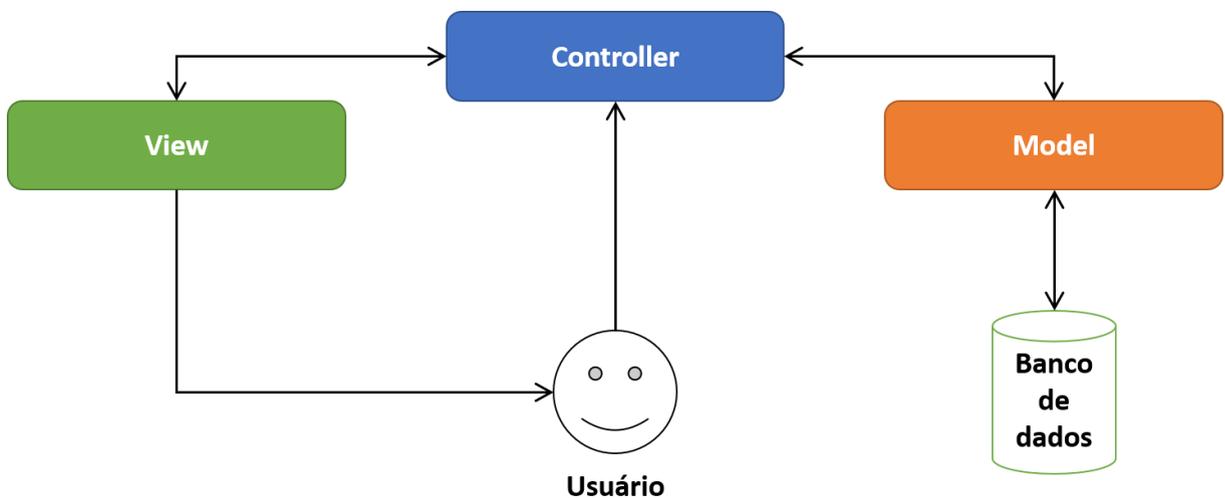
4.3 Desenvolvimento

4.3.1 Arquitetura MVC

Model-View-Control ou mais conhecido como MVC, foi desenvolvido para poder sanar um problema encontrado do qual seria controlar um grande e complexo conjunto de dados de forma simultânea, inicialmente desenvolvido por Trygve ReensKaug no ano de 1978 para ser empregada pela linguagem Smalltalk (CUSTODIO, 2012).

Atualmente o modelo MVC tem sido o padrão mais utilizado e difundido no meio do desenvolvimento de software por ser simples e de que todas linguagens dão um suporte para o mesmo. O MVC funciona dividindo responsabilidades, sendo que cada sílaba representa sua respectiva responsabilidade, sendo o M de Modelo, onde tem-se uma classe da qual representa o objeto do qual gostaríamos, como um modelo de produto ou de usuário. O V de Visão, onde seria a parte visível do software e C de Controle onde estaria localizada a regra de negócio (CELESTINO, 2014).

Figura 5 – Exemplo do MVC.



Fonte: www.treinaweb.com.br.

4.3.2 Flutter

O Flutter é um kit de ferramentas da Google para poder construir aplicações de uma forma mais rápida e seguindo o padrão de estilo do Material Tema, sendo assim podendo ser compilado nativamente para múltiplos ambientes, tanto para mobile, web ou até mesmo desktop a partir de um único código fonte (MELO, 2019).

Atualmente o Flutter está baseado na linguagem Dart, a qual também foi desenvolvida pela Google. Flutter foi desenvolvido para tentar repassar uma sensação melhor para os desenvolvedores, pois muitas vezes a Google tinha que ter times para desenvolver nativamente para IOS(Sistema operacional utilizado em celulares da marca Apple) e Android, do qual acabava dificultando a manutenção e implementação de novas funcionalidades nos softwares da empresa (MELO, 2019).

O Flutter se destacou bastante nas comunidades de multi-plataforma pela forma como gerencia a renderização dentro dos sistemas operacionais. Está dividido em 3 camadas na sua engine, sendo a primeira responsável por todo o kit de Flutter o qual é escrito em Dart, ou seja onde o desenvolvedor estará escrevendo o código. A segunda sendo o core do Flutter onde irá ocorrer todo o processamento de tratamento de ações e por último sua engine gráfica própria que é escrita em OpenGL. Esta é a diferença do Flutter para outros kits, pois os kits de desenvolvimento anteriores a ele, utilizaram APIS(Application Programming Interfaces, interface de programação de aplicações são comumente utilizadas para prover serviços a terceiros) do próprio sistema operacional para fazer renderização, com o Flutter ele mesmo se responsabiliza por esse processo acabando por torná-lo performaticamente superior (MAGALHÃES, 2019).

4.3.3 Node JS

Node.js pode ser considerado como um ambiente para executar Javascript sem precisar de um browser, como normalmente é utilizado. O Node.js foi desenvolvido por um grupo de desenvolvedores contratados pela Google para construção da V8, engine atual utilizada dentro do Google Chrome, do qual aumentou o desempenho de execução do javascript dando a possibilidade de usar como um ambiente de execução de aplicações diversas (LENON, 2018).

A principal característica e que diferencia ele das outras tecnologias está relacionado a maneira como ele gerencia sua parte de thread, enquanto outras linguagens como PHP e Java trabalham com multi-threads o Node.js utiliza uma única thread (LENON, 2018).

Quando modelos tradicionais recebem requisições os mesmos irão criar thread para cada requisição, e como o servidor possivelmente vai ter um limitação de hardware, quando atingir o limite do hardware o mesmo fará as requisições novas esperar até que se libere novas threads, fazendo assim ter muitas vezes um gargalo na velocidade das requisições. O Node.js trabalha com um conceito de não-bloqueante, ou seja, toda vez que ele precisa algo ele não bloqueia a thread fazendo assim um efeito de que tem multi-processos rodando de forma paralela (LENON, 2018).

4.3.4 Flask

Flask é um Micro-Framework desenvolvido em cima da linguagem Python, no ano de 2010, para ser uma forma mais simples de desenvolvimento web para Python, pois até o momento de seu lançamento o ambiente de desenvolvimento web para Python era limitado somente ao framework Django, do qual precisa de conhecimentos mais profundos na web para a utilização (ANDRADE, 2019).

Por Flask ser um Micro-Framework o mesmo acaba sendo muito modularizado, deixando a cargo do desenvolvedor muitas escolhas, e por tal motivo seu uso quase sempre está ligado a criação de microsserviços e apis (ANDRADE, 2019).

Flask se destaca entre os framework web na comunidade Python, pois tem uma simplicidade grande, onde muitas vezes é possível fazer boa parte do desenvolvimento em um único arquivo e conforme a necessidade pode-se migrar a forma como está estruturado o projeto de maneira rápida e simples. O Flask provê rapidez no desenvolvimento, pois necessita de pouca configuração e, conforme necessitar de algo, é possível adicionar de forma simples, então acaba se tornando muito focado em produtividade (ANDRADE, 2019).

4.3.5 MySQL

O MySQL foi criado na Suécia por um grupo de dois suecos e um finlandês que são David Axmark, Allan Larsson e Michael “Monty” Widenius, que trabalham juntos desde da década de 1980. Atualmente todo o contexto de desenvolvimento e manutenção do MySQL envolve uma equipe de aproximadamente 400 funcionários espalhados pelo mundo (OFICINA, 2010).

O MySQL é um sistema de gerenciamento de banco de dados, também conhecido como SGBD, foi desenvolvido para ser utilizado com a linguagem de SQL (Standard Query Language - Linguagem de Consulta Estruturada). No qual é considerado um dos banco de dados mais utilizados no mundo inteiro com uma média de 10 milhões de instalações pelo mundo (OFICINA, 2010).

4.3.6 Ambiente de desenvolvimento

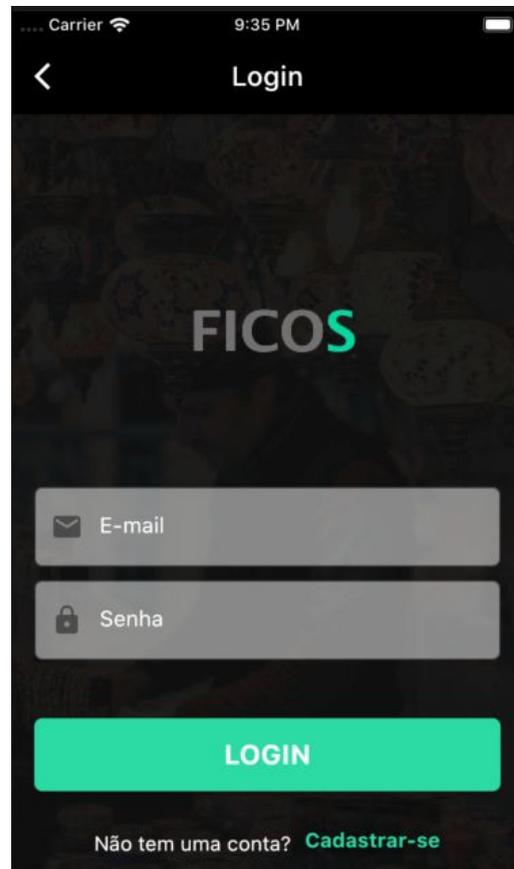
Como ambientes de codificação foram utilizados os programas desenvolvidos pela JetBrains, como o PyCharm para desenvolvimento da apis em Flask, WebStorm para desenvolvimento da parte do servidor do chat que utilizou o Node.js, IntelliJ IDEA para desenvolvimento do aplicativo e o DataGrip para consultas no banco de dados.

No desenvolvimento do projeto foi utilizada a versão 20.04 do Ubuntu 64 bits, com o uso do git para subir e manter o versionamento de código, salvando todo o versionamento no GitHub.

5 RESULTADO

O projeto do aplicativo FiCos, teve como finalidade a criação de um aplicativo para ajudar clientes e vendedores locais a se conhecerem, possibilitando assim a comerciantes locais conseguirem manter uma concorrência com empresas com grande marketing na web e também aproximando o cliente de estabelecimentos locais, que possam suprir suas necessidades , mas como não tinham acesso, muitas vezes procuravam por lojas maiores.

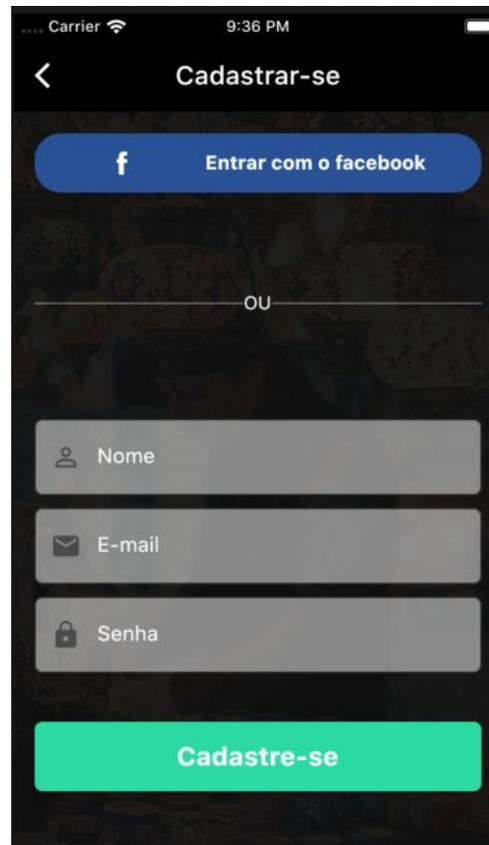
Figura 6 – Tela de Login do App FiCos



5.1 Tela de Login

Na tela inicial do app, o usuário tem a opção de fazer login, para poder enviar mensagem e também adicionar itens para venda, caso o usuário não tenha uma conta previamente cadastrada, no rodapé da aplicação tem a opção de ser redirecionado para a tela de cadastro. Para o usuário acessar a aplicação com seu usuário, precisa informar seu e-mail e senha para acessar, caso aconteça algum erro durante o login, o aplicativo irá informar com uma mensagem em cima do botão, mas se tudo ocorrer de forma correta, o mesmo será direcionado para a tela inicial do FiCos.

Figura 7 – Tela de Cadastro do App FiCos.



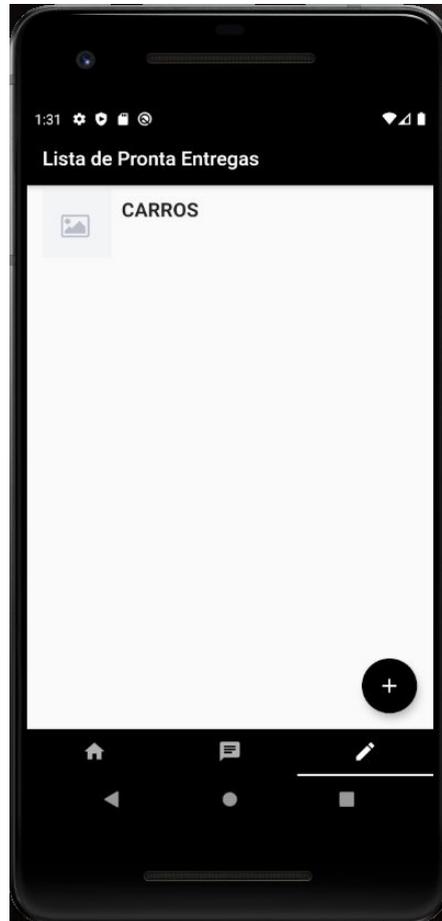
Fonte: Autor.

5.2 Tela de Cadastro

A tela do cadastro do FiCos é dividida em duas, na parte superior aparece a opção de fazer o login utilizando a conta pessoal do Facebook, onde o usuário será direcionado ao aplicativo do próprio Facebook, para poder dar acesso aos dados do mesmo e, após confirmar, retornará ao FiCos sendo redirecionado para a tela inicial do aplicativo.

A segunda parte da tela contém os campos para cadastro diretamente do aplicativo, onde o usuário deve informar um nome, o qual será mostrado nos seus anúncios e quando iniciar uma conversa no chat, seu e-mail pessoal, sendo que caso informe um endereço já cadastrado no aplicativo o mesmo será informado que não pode cadastrar um e-mail duplicado, e uma senha. Quando o usuário efetuar o login, será necessário informar a senha cadastrada juntamente com seu e-mail.

Figura 8 Lista de pronta entrega.



Fonte: Autor.

5.3 Pronta Entrega

5.3.1 Lista de Pronta Entrega

Após o usuário realizar o login na ferramenta, perceberá que os ícones de mensagem e de edição de pronta entrega estarão habilitados, ao clicar no ícone de pronta entrega, o mesmo se depara com a suas listas de pronta entregas cadastradas.

Caso o usuário deseje deletar uma pronta entrega cadastrada, deve clicar em cima da pronta entrega escolhida e arrastar para a direita, onde poderá notar um ícone de lixeira e perceberá que a pronta entrega e seus itens foram deletados da sua listagem.

Se o usuário desejar editar o nome, mudar a distância de alcance da pronta entrega ou adicionar itens, deve clicar em cima da pronta entrega e arrastar para esquerda, ao executar o movimento perceberá que irá aparecer um ícone de edição e será direcionado para a tela de edição da pronta entrega.

Figura 9 - Tela Criação de Pronta Entrega.

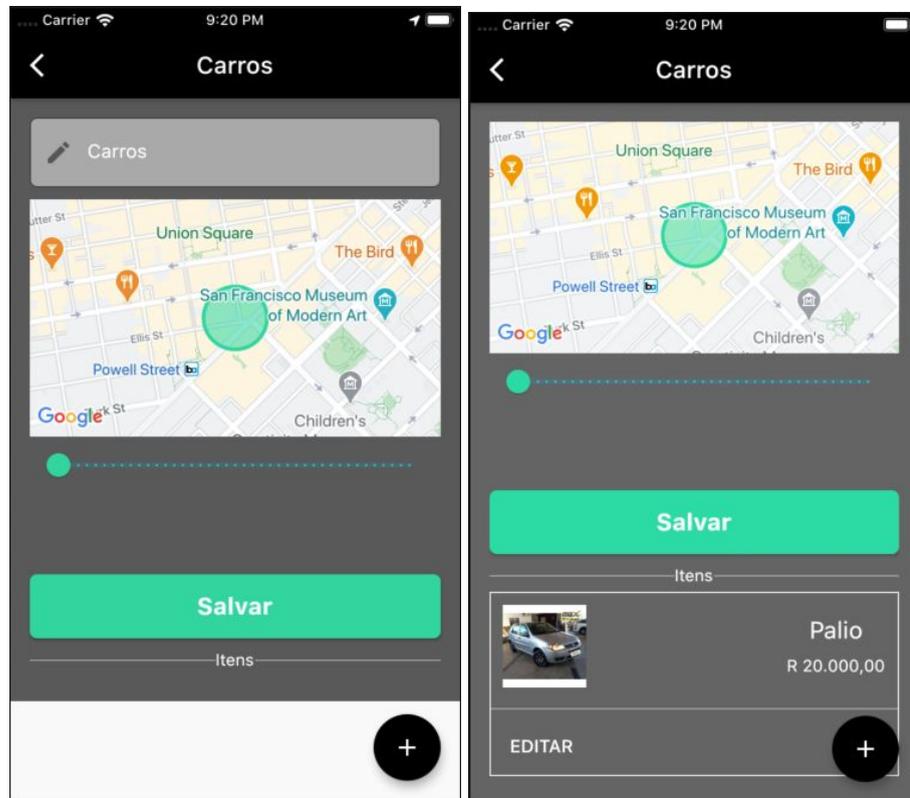


Fonte: Autor.

5.3.2 Tela Criação de Pronta Entrega

Na tela anterior é possível observar o botão de adicionar uma pronta entrega, ao clicar, ele direciona para a tela de cadastro de uma pronta entrega, onde o usuário deve preencher o nome e clicar em salvar.

Figura 10 – Tela de Edição de Pronta Entrega.



Fonte: Autor.

5.3.3 Tela de Edição da Pronta Entrega

Após o usuário criar a pronta entrega, ao clicar para editar, será direcionado para a tela de edição da pronta entrega. Nessa tela, poderá editar o nome da pronta entrega, ajustar o raio de entrega dos itens dentro da pronta entrega, deslizando o ícone na linha embaixo do mapa e conforme for movimentado será mostrado o raio de ação dentro do mapa. Para poder pegar o posicionamento foi utilizada a API do Google Maps, que permite triangular a posição do dispositivo. Terá a listagem dos itens da pronta entrega embaixo das configurações da pronta entrega e o botão para poder adicionar mais itens na pronta entrega.

Figura 11 – Tela de Criação e Edição dos Itens.

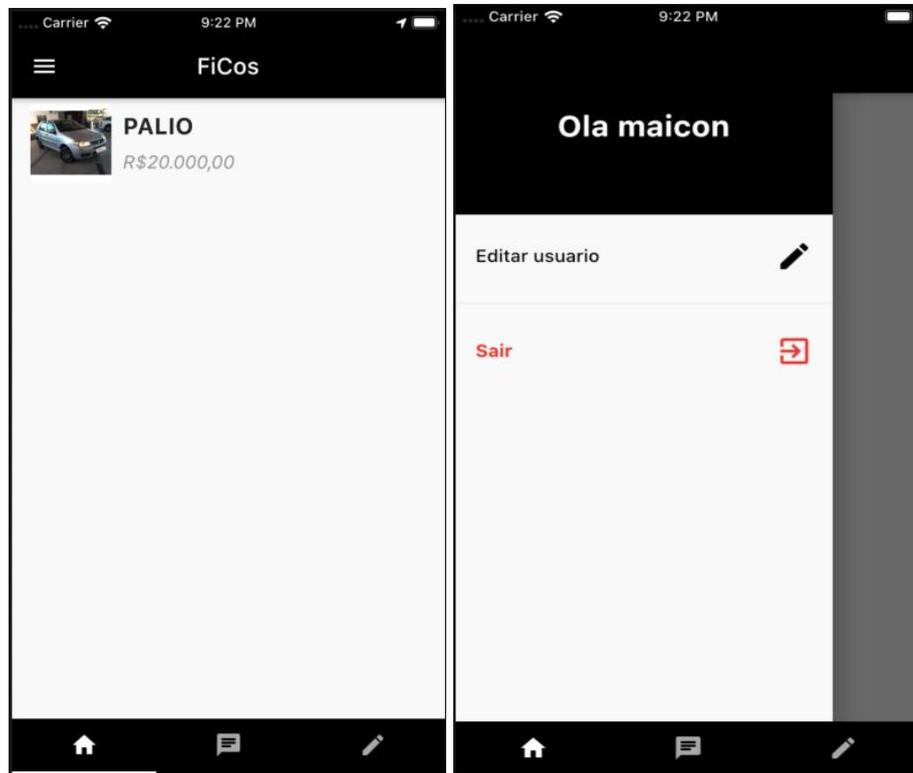


Fonte: Autor.

5.4 Itens

5.4.1 Tela de criação e edição de item da pronta entrega

Nesta tela o vendedor vai poder realizar o cadastro de itens da pronta entrega, onde é possível adicionar imagens, num total de no máximo 6 imagens, informar o nome do item, o valor que pretende cobrar, uma descrição do que seria o item, um menu onde o vendedor escolhe em qual categoria o item se encaixa e, após preencher todos esses dados, pode salvar o item. Ao selecionar um item para editar, será direcionado para essa mesma tela.

Figura 12 – Catálogo.

Fonte: Autor.

5.5 Catálogo

5.5.1 Tela de Catálogo

A tela inicial do aplicativo, que será mostrada ao usuário ao iniciar o aplicativo, por mais que não tenha uma conta cadastrada, vai ser a tela de catálogo, onde serão mostrados os últimos itens cadastrados que são entregues e disponibilizados na sua região. Para poder fazer essa triangulação foi utilizada a API do Google Maps para pegar a posição do dispositivo do qual informa o back-end e através da longitude e latitude, o banco de dados realiza a consulta levando em consideração esses dois parâmetros. O usuário pode clicar no ícone à direita para filtrar os itens por categoria, ou pode fazer uma busca através do campo que se localiza no topo da página. Ao clicar no menu hambúrguer, o usuário terá a opção de fazer login ou logoff da aplicação.

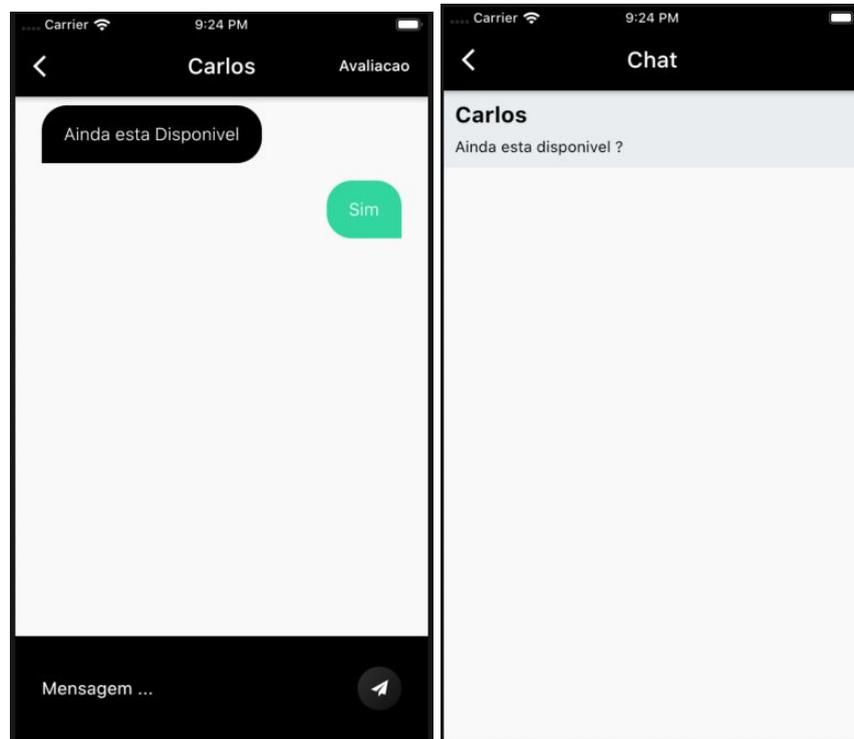
Figura 13 – Tela Detalhe do Produto.



Fonte: Autor.

5.5.2 Tela Detalhe do Produto

Após o usuário clicar em algum produto na tela inicial, será direcionado para a tela onde irá conter mais detalhes do produto, como todas as imagens cadastradas pelo vendedor, a descrição completa do produto e logo abaixo o cliente poderá ver o botão para iniciar o chat.

Figura 14 – Tela de Chat.

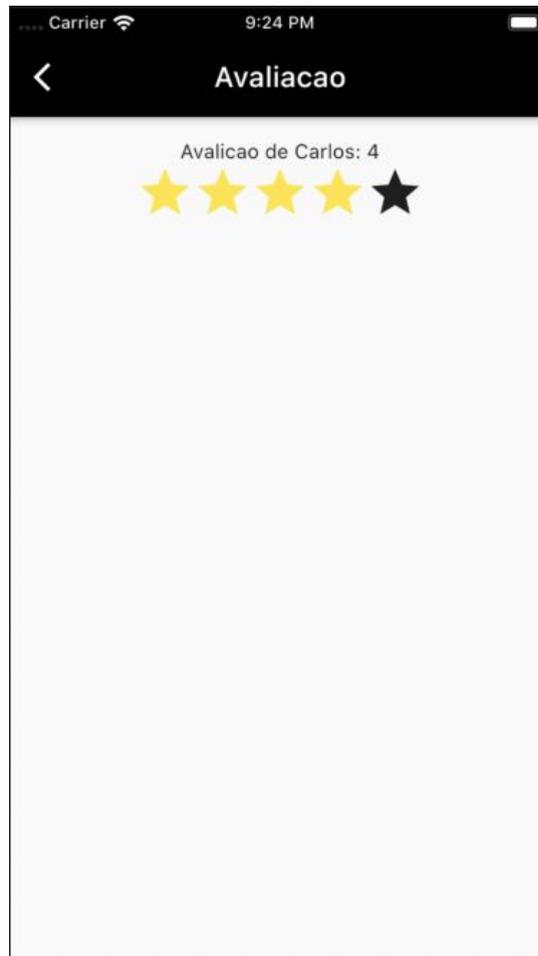
Fonte: Autor.

5.6 Chat

5.6.1 Chat

Se o cliente clicar no chat, será direcionado para a tela de chat, onde o cliente poderá ver um campo onde pode inserir o texto que deseja para iniciar uma conversa ou responder o cliente. Ao enviar a mensagem, o vendedor irá receber a notificação que tem uma nova mensagem e poderá responder. A tela de chat contém um menu no lado direito onde tem a opção de avaliação tanto de cliente para vendedor quanto de vendedor para cliente.

Figura 15 – Tela Avaliação de Cliente ou Vendedor.



Fonte: Autor.

5.7 Avaliação de Cliente ou Vendedor

5.7.1 Tela de Avaliação de Cliente ou Vendedor

Caso o cliente ou o vendedor desejar realizar uma avaliação após ter uma conversa com, no mínimo, 5 mensagens de ambos os lados, terá a opção de avaliação liberada no menu do chat. Ao selecionar a avaliação será direcionado para a tela onde conterà o nome de quem será avaliado e 5 estrelas, onde o avaliador poderá selecionar entre as 5 estrelas marcando o máximo que ele achou que foi respondido e atendido.

Figura 16 – Tela de Visualização de Avaliação.



Fonte: Autor.

5.7.2 Tela de Visualização de Avaliação.

Caso o cliente queira ver como o vendedor foi avaliado por outros clientes, poderá clicar em um botão no menu do lado direito da tela de detalhe do produto. Ao clicar, o cliente será direcionado para uma tela onde conterà o total de avaliações e a média de estrelas que a maioria dos clientes informaram para aquele vendedor.

Caso o vendedor deseje ver a avaliação que o cliente recebeu de outros vendedores, na conversa terá um menu na direita onde conterà uma opção de visualizar as avaliações e, ao selecionar, será direcionado para a tela que mostra as avaliações antigas.

6 TESTES

Os testes sobre a aplicação foram elaborados e executados usando quatro etapas diferentes entre si, utilizando o princípio do método da caixa preta, da qual permite gerenciar uma série de condições de entrada que irão utilizar todos os requisitos do programa apresentado. Esse teste foi desenvolvido para tentar encontrar erros de funções incorretas, erros de interface, desempenho de inicialização ou término e de estrutura de dados (PRESSMAN, 2011). Foram criados seis usuários, três do tipo cliente e outros três usuários do tipo vendedor. Para o papel dos vendedores, foram criados três anúncios contendo dados falsos, e esses anúncios foram escolhidos de forma aleatória, sendo cada anúncio com dados totalmente diferentes. Para os clientes foram escolhidos dados falsos também e totalmente diferentes.

Na primeira parte dos testes, a cada finalização de uma tarefa, tanto com o desenvolvimento do back-end ou do front-end, a tarefa era submetida a vários testes, como não realizar o que era esperado, informando dados errados e etc, para verificar se as funcionalidades implementadas estavam funcionando da forma como era esperada, caso a funcionalidade apresentasse alguma falha, voltava-se para o desenvolvimento para fazer a correção do problema e era novamente submetido a bateria de testes.

Na segunda parte dos testes, foram realizados os testes onde estão as tarefas interligadas, por exemplo, para criar uma pronta entrega deve-se ter uma conta registrada. A cada necessidade de ajustes por apresentar algo que não era o comportamento esperado, era revisto toda a cadeia de relação entre as tarefas e observado onde deveria ser corrigido. Após esse processo, era enviado ao desenvolvimento novamente para as correções necessárias e, após o término das correções, testado todo o vínculo da tarefa novamente para verificar se o problema foi sanado.

A terceira fase, se direcionou a realização de testes da aplicação como um todo, onde foi testado todo o fluxo de uso da ferramenta, sendo o fluxo completo para os seis usuários criados. Como descrito na etapa anterior, caso houvesse um comportamento diferente do esperado, voltava-se para a fase de desenvolvimento e, após as correções realizadas, se repetia o teste.

Após a conclusão do ciclo de desenvolvimento e testes primários direcionados ao desenvolvimento do sistema, realizou-se alguns testes de usabilidade com alguns usuários reais, testando os dois perfis permitidos no sistema. Através dos retornos obtidos dos usuários, serão montadas atualizações futuras para o sistema.

7 CONCLUSÃO

Este artigo dissertou sobre a elaboração do aplicativo FiCos. Com a separação do software por perfil de vendedor e de cliente, tem como objetivo principal ajudar comerciantes locais a ofertarem seus produtos a clientes próximos, trazendo assim a facilidade de divulgação para os mesmos, como uma forma de ajudar o comércio local, aproximando clientes de estabelecimentos próximos do qual não tinham conhecimento prévio, fornecendo uma possibilidade do comerciante local ter uma disputa mais igual com redes varejistas maiores.

Empregando o padrão MVC o desenvolvimento da aplicação foi construído através do Framework Flutter, que faz a utilização da linguagem Dart desenvolvida pela Google. Sendo um dos Framework mais populares para o desenvolvimento de aplicativos para multiplataformas, por seu processo de renderização. Com auxílio de componentes do Flutter e de Api's, principalmente da Google, foi possível fazer a integração com mapas, localizações e distâncias de pontos. Para a integração de dados da Google com a aplicação e principalmente para gerenciamento de dados, foi desenvolvida uma Api com Flask, uma biblioteca para utilização do Python para web, onde foi utilizada junto com o banco de dados MySQL para persistência de dados da aplicação. Também foi desenvolvida uma Api para uso do chat que existe na aplicação, utilizando o Node.js.

Com o aumento do uso do FiCos, pela adesão de comerciantes e por clientes, o mesmo pode vir a ajudar nas vendas desses comerciantes, colaborando com uma forma de concorrência direta com as redes varejistas que estão presentes na internet e consecutivamente irá ajudar clientes a descobrirem mais o comércio local e através disso aumentando o dinheiro em circulação e o número de empregos locais.

REFERÊNCIAS

ALVES, Júlio. **Como funciona o React Native**. Disponível em:

<https://www.housecursos.com/blog/como-funciona-react-native/>. Acesso em: 15 mar. 2019.

ANDRADE, Ana Paula. **O que é Flask?**. Disponível em:

<https://www.treinaweb.com.br/blog/o-que-e-flask/>. Acesso em: 06 outubro. 2020.

CALSAVARA, Alcides. **Modelagem de Software Orientado a Objetos utilizando a Linguagem UML**. Disponível em:

<http://www.ppgia.pucpr.br/~alcides/Teaching/ProgramasAprendizagem/ModelagemOrientadaObjetos/Oo.zip>. Acesso em: 03 outubro. 2020.

CUSTODIO, Glaucio. **Desenvolvimento em Camadas**. Disponível em:

<https://glaucocustodio.github.io/2012/08/09/desenvolvimento-em-camadas-com-mvc/>. Acesso em: 04 outubro. 2020.

CELESTINO, Andre Luis. **Conceito e as dúvidas sobre o MVC**. Disponível em:

<https://www.profissionaisti.com.br/o-conceito-e-as-duvidas-sobre-o-mvc/>. Acesso em: 04 outubro. 2020.

GUEDES, Gilleanes T. A. **UML 2 Uma Abordagem Prática**, 2ª Edição, Novatec, São Paulo, 2011.

INGIZZA, Carolina. **Vendas online no Brasil crescem 47% no 1º semestre, maior alta em 20 anos**. Disponível em:

<https://exame.com/pme/e-commerce-brasil-cresce-47-primeiro-semester-alta-20-anos/>. Acesso em: 30 setembro 2020.

LENON. **Node.js - O que é, como funciona e vantagens**. Disponível em:

<https://www.opus-software.com.br/node-js/>. Acesso em: 06 outubro. 2020.

MELO, Rubens. **O que é Flutter?**. Disponível em:

<https://www.flutterparainiciantes.com.br/o-que-e-flutter>. Acesso em: 05 outubro. 2020.

MAGALHÃES, Tulio. **Flutter: tudo sobre o queridinho do google**. Disponível em:

<https://www.zup.com.br/blog/flutter>. Acesso em: 05 outubro. 2020.

PAREDES, Arthur. **O que a metodologia Kanban e como utilizar**. Disponível em:

<https://www.iebschool.com/pt-br/blog/empreendedores-e-gestao-empresarial/agile-e-scrum/o-que-e-a-metodologia-kanban-e-como-utiliza-la/>. Acesso em: 01 outubro. 2020.

PEREIRA, Alex. **Metodologia Ágil: Entenda o que é e quais são as 8 mais utilizadas**.

Disponível em: <https://neilpatel.com/br/blog/metodologia-agil/>. Acesso em: 01 outubro. 2020.

PRESSMAN, Roger S. **Engenharia de Software Uma Abordagem Profissional**, 7ª Edição, McGraw-Hill Companies, New York, 2011.

OFICINA, Redacao. **O que é MySQL?**. Disponível em: https://www.oficinadanet.com.br/artigo/2227/mysql_-_o_que_e. Acesso em: 06 outubro. 2020.

RIVEIRA, Carolina. **Como a alta informalidade no Brasil freia a produtividade das empresas**. Disponível em: <https://exame.com/negocios/como-a-alta-informalidade-no-brasil-pode-frear-a-produtividade/>. Acesso em: 30 setem. 2020.

ROCHA, Thiago. **Como o Marketing pode se adaptar ao impacto do coronavírus para que sua empresa saia fortalecida da crise**. Disponível em: <https://resultadosdigitais.com.br/blog/marketing-em-tempos-de-coronavirus/>. Acesso em: 30 setem. 2020.

RONDINELLI, Julia. **Cresce a importância do marketplace no comércio eletrônico**. Disponível em: <https://www.ecommercebrasil.com.br/noticias/marketplace-no-brasil-ebit/>. Acesso em: 01 outubro. 2020.

SPONCHIADO, Ricardo. **7 Vantagens de vender em um marketplace**. Disponível em: <https://www.ecommercebrasil.com.br/artigos/7-vantagens-de-vender-em-um-marketplace/>. Acesso em: 01 outubro. 2020.

SAMPAIO, Daniel. **Marketplace: o que é e como alavancar as vendas por meio de uma plataforma online**. Disponível em: <https://rockcontent.com/br/blog/marketplace/>. Acesso em: 01 outubro. 2020.

ZACHO, Ricardo. **O que é Marketplace? - Veja as vantagens e desvantagens**. Disponível em: <https://www.ecommercebrasil.com.br/artigos/marketplace-vantagens-e-desvantagens/>. Acesso em: 01 outubro. 2020.