

FretExpresso: Software WEB com integração da API Google Maps para solicitação de fretes

Julio Cesar Leite de Jesus¹, Leandro Camara Ledel²

¹Curso de Tecnologia em Análise e Desenvolvimento de Sistemas – Campus Hortolândia – Instituto Federal de São Paulo (IFSP)

²Área de Informática – Câmpus Hortolândia _ Instituto Federal de São Paulo (IFSP)

julio.leite@aluno.ifsp.edu.br, ledel@ifsp.edu.br

Abstract. *This scientific article presents web software for requesting freight, using Vue.JS, integration with Google Maps API and use of Firebase tools, such as Cloud Firestore and Authentication. The software was developed with the aim of facilitating freight requests for people who chose to work in the sector due to increased unemployment during the 2020 pandemic. The software was developed using the Vue.JS framework, which allows the creation of user interfaces of quickly and efficiently. Integration with the Google Maps API allows the user to view the freight route and the estimated cost of the ride. Using Cloud Firestore tools allows data to be saved in JSON format and Authentication allows the user to create an account in the system and log in to request shipments*

Resumo. *O presente artigo científico apresenta um software web para solicitação de fretes, com uso de Vue.JS, integração com API do Google Maps e uso de ferramentas do Firebase, como Cloud Firestore e Authentication. O software foi desenvolvido com o objetivo de facilitar a solicitação de fretes para pessoas que optaram por trabalhar no ramo devido ao desemprego aumentado na pandemia de 2020. O software foi desenvolvido utilizando o framework Vue.JS, que permite a criação de interfaces de usuário de forma rápida e eficiente. A integração com a API do Google Maps permite que o usuário visualize o trajeto do frete e o valor estimado da corrida. O uso das ferramentas do Cloud Firestore permite que os dados sejam salvos no formato JSON e o Authentication permite que o usuário crie uma conta no sistema e faça o login para solicitar fretes*

1. Introdução

O desemprego vem tomando conta de parte da população brasileira. Isso foi constatado em uma pesquisa do IBGE do primeiro trimestre de 2020, a qual identificou que a taxa de desemprego no país estava em 12,6%, mostrando um cenário preocupante com relação à falta de oportunidades de emprego. Podemos relacionar o fato desses números como consequência da COVID-19, que devastou o mundo e fez com que perdessem seus empregos. (IBGE, 2020).

Como maneira de superar esse cenário, uma parcela da população que possuem veículos próprios, recorreram a realizar pequenos trabalhos de movimentação de cargas e pessoas, conforme constatado pela Pesquisa Nacional de Amostra a Domicílios (PNAD), que apontou que o aumento destes serviços entre 2012 e 2019 foi de 137,2%.

Normalmente a divulgação desse tipo de trabalho ocorre de maneira informal e/ou divulgação em redes sociais. No caso de transporte de pessoas, há a presença de aplicativos de transporte que facilitam a centralização dos chamados. Já no caso chamado transporte de pequenas cargas se trata de solicitações para transporte de móveis em mudanças, como sofás, geladeiras, camas, entre outras coisas.

Normalmente a pessoa interessada procura com familiares e amigos próximos indicações de profissionais que realizam esse tipo de serviço, informando a necessidade e o local de partida e destino da mudança. Outro meio de encontrar esse tipo de serviço é através de redes sociais como *Facebook*, através da ferramenta *Marketplace*, onde os profissionais podem oferecer o seu serviço, expondo o que ele atende e qual o seu telefone de contato.

Dado esse cenário, o trabalho tem como objetivo propor um sistema *web* para solicitação de serviços de frete, e ser um meio de gerenciamento dessas viagens para os motoristas. A intenção é a centralização de solicitações, com a presença de mapas, cálculos de corridas e *chat* para facilitar a interação entre motorista e o solicitante.

2. Trabalhos Correlatos

Para se obter uma melhor visão do contexto de aplicação e da necessidade do *software* de fretes, foi realizada uma pesquisa, com o buscador padrão Google, para obter informações como: até que ponto uma pesquisa desse tema foi realizada? Há *softwares* com a mesma proposta atualmente no mercado? Se sim, qual o diferencial?

A presente Seção apresenta os dois trabalhos correlatos encontrados.

2.1 Sistema ControLOG

Durante as pesquisas, foi encontrado um *software web*, denominado ControLOG.

O Controlog é uma plataforma, na nuvem, para controle de entregas e coletas de encomendas que, aliado a um aplicativo para celular Android, permite o acompanhamento da operação logística em tempo real (CONTROLOG, 2024).

O plano gratuito é limitado a dois entregadores. Depois os planos de pagamento vão ficando mais completos, porém mais caros, sendo que o plano “Profissional 10” – topo de linha - custa R\$ 1125,48 reais/mês.

2.2 *Uber Flash*

A *Uber Flash* surgiu por volta de maio de 2020, lançado pela empresa *Uber*. O serviço tem o propósito de realizar pequenas entregas, solicitadas via aplicativo, de objetos que possuam no máximo 20 kg, segundo próprio *site* da *Uber* e que possam ser acomodados com segurança no porta-malas do veículo.

2.3 Proposta do FretExpresso

O *software web* FretExpresso tem como diferencial ser voltado tanto a usuários finais quanto aos motoristas. No caso dos usuários finais, eles possam solicitar uma corrida de frete instantaneamente. Já os motoristas podem aceitar e controlar de maneira mais eficiente as viagens realizadas no dia/mês/ano.

A ideia do FretExpresso é ser disponibilizado de forma gratuita, semelhante ao Uber, o que o diferencia do ControLog, por exemplo.

Comparando-se o Uber Flash com o *FretExpresso*, o que os diferencia principalmente é a capacidade e limite de peso entre as cargas, onde na solicitação de frete, esse último permite ultrapassar 20 kg desde que seja realizado em veículos os quais possuam a carroceria apropriada.

Abaixo temos a tabela comparativa entre os dois trabalhos correlatos com a FretExpresso de acordo com dois atributos: **Custo e Limite de Peso**

Atributos	ControLOG	Uber Flash	FretExpresso
Custo	Grátis até 2 motoristas	Grátis	Grátis
Limite de pesos	Não há	Até 20kg	Não há

Tabela 1: Comparativo entre trabalhos correlatos.

3. Referencial Teórico

Nessa seção são apresentados alguns conceitos essenciais para o desenvolvimento do projeto, tratando-se de definições, linguagem de programação, metodologia adotada e ambiente de desenvolvimento.

3.1 *Desenvolvimento Web*

O desenvolvimento na linguagem *web* vem se tornando uma coisa comum, já que se trata de uma tecnologia bastante abrangente e que pode ser acessada por *smartphones*, *smartwatches*, *notebook*, *tablets* em uma mesma residência por exemplo. Por esse motivo, para garantir maior mobilidade aos usuários quanto ao motorista, essa foi a tecnologia escolhida para fazer parte do desenvolvimento do projeto.

Algumas vantagens da utilização dessa tecnologia são destacadas no trabalho de Camargo (2010), que cita trecho do livro de Noergaard: “Quando a escrita do projeto é realizada seguindo padrões, é possível que ela seja modificada sem que haja a necessidade de aplicações complexas para isso”.

Já como desvantagem foi detectada a conectividade por exemplo, já que as aplicações *web* dependem de uma conexão ativa e preferencialmente de boa qualidade, para que possam garantir um bom desempenho e obter os dados de maneira mais eficiente.

3.2 *Firebase, Cloud Firestore e Firebase Authentication*

O *Firebase* se trata de uma plataforma da *Google Cloud Platform* para desenvolvedores que fornece a infraestrutura de *backend*¹ necessária para desenvolvedores que desejam focar na parte de negócio da aplicação (FIREBASE, 2024 a).

¹ Código responsável à conexão com servidores, banco de dados, segurança, estrutura, gerenciamento de conteúdo e atualizações. (EWALLY)

Já o *Cloud Firestore* consiste em um banco de dados não relacional (*NOSQL*) hospedado em nuvem, onde os dados são armazenados no formato *JSON*² e sincronizado em tempo real para clientes conectados. (FIREBASE, 2024 b).

O *Firebase Authentication* fornece bibliotecas e *SDKs*³ para autenticação de usuários nos aplicativos (FIREBASE, 2024 c).

3.3 API Google Maps

Segundo Erle e Gibson (2006) a utilização do *Google Maps* é simples e interessante, pois é capaz de exibir um mapa no *website*, permitindo uma personalização por parte do desenvolvedor de acordo com as suas necessidades.

O uso da *API* do *Google Maps* se dará no trabalho à medida que será necessária para fazer a apresentação e localização dos pontos de partidas e início de uma solicitação de frete.

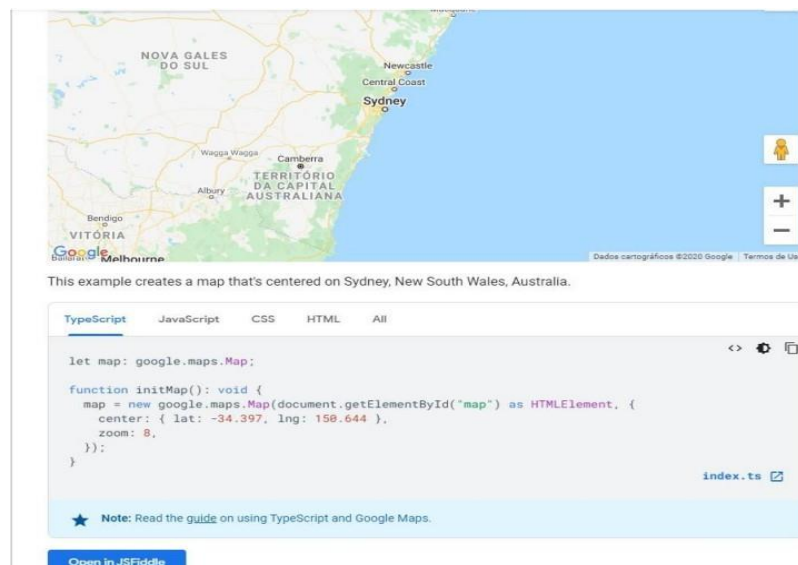


Figura 1: Exemplo de implementação *API Google Maps*.

3.3.1 API Distance Matrix

A *API Distance Matrix* do Google Maps fornece informações de acordo com a rota especificada de origem e destino (que pode ser na forma de Ids de lugar, endereço ou

² Formatação utilizada para estruturar dados em formatos de texto e transmiti-los de um sistema para outro. (DE SOUZA, 2020)

³ Conjunto de ferramentas de criação específicas da plataforma para desenvolvedores. (AMAZON)

coordenadas de latitude/longitude) com base no meio de transporte escolhido (carro, bicicleta, transporte público ou a pé).

Através de uma solicitação *HTTP*, a *API* é capaz de retornar em formato *JSON* ou *XML*⁴ informações importantes para a rota como: Distância e tempo de viagem entre a origem e o destino (GOOGLE, 2024).

3.3.2 API Maps Javascript

A *API Maps Javascript* do Google Maps permite adicionar mapas personalizados e dinâmicos em páginas da *web* e dispositivos móveis.

Para essa *API* é possível selecionar o modo de mapa a ser exibido: roteiro, satélite, híbrido ou terreno (GOOGLE, 2024).

3.3.3 API Places

A *API Places* do Google Maps contém uma série de recursos úteis que podem ser utilizados, como por exemplo o *Place Search* e o *Place Autocomplete*.

O *Place Search* retorna uma lista de lugares baseado na localização do usuário ou de um endereço específico.

Já o *Place autocomplete* identifica o endereço que está sendo digitado pelo usuário e o autocompleta, facilitando assim a busca por um endereço, além de outras opções úteis (GOOGLE, 2024).

3.4 Vue.js

O *Vue.js* se trata de um *framework Javascript Open Source* que fornece estrutura para construção de interfaces de usuário via componentes (DOUTBOX, 2023).

O *Vue.js* se baseia em *HTML*, *CSS* e *Javascript* e permite a criação de aplicações reativas.

⁴ Linguagem de marcação que fornece regras para definir quaisquer dados para intercâmbio de informações entre sistemas de computador como sites, banco de dados e aplicações de terceiros. (AMAZON)

Entre suas principais vantagens podemos citar a facilidade em criar partes individuais da interface de maneira isolada, facilidade de aprendizado, por se tratar de uma junção entre *HTML*, *CSS* e *Javascript* e a reatividade e interatividade, onde permite que as alterações realizadas em seu código fonte sejam refletidas de maneira mais rápida para o usuário (DOUTBOX, 2023).

3.5 Javascript

Criada pela *Netscape*, *Javascript* se trata de uma linguagem de programação *web* para auxiliar na criação de *websites*, sendo ela responsável por determinar o comportamento de uma página *web*.

Segundo Flanagan (2013), o *Javascript* permite a manipulação do *HTML* e do *CSS*, além de definir o comportamento através das chamadas rotinas de tratamento de eventos, que aguardam determinada ação ocorrer para executar algum comando.

Alguns dos eventos podem ser: um clique em determinado local da página, quando a página termina de carregar; ou através do pressionamento de uma tecla.

3.6 HTML

Hyper Text Markup Language (*HTML* – Linguagem de marcação de HiperTexto) permite que seja criada a estrutura básica de um site de internet através de Tags. Com um funcionamento de uma linguagem interpretada, ela é editada e exibida pelos navegadores, permitindo o acesso e interação do usuário com o *site*. (DIO, 2023)

3.7 CSS

Cascading Style Sheets (*CSS* – Folhas de estilo em cascata) é uma tecnologia utilizada em conjunto com o *HTML* para manipulação da parte visual do *website*, onde é especificado fontes, cores, margens, bordas, imagens de fundo, alinhamento, posicionamento e outras propriedades de cada elemento presente na página.

4. Metodologia

O trabalho seguirá a metodologia do modelo incremental, considerado por Pressman (2011) como: “[,,] processo incremental tem seu foco voltado para a entrega de um produto operacional com cada incremento. Os primeiros incrementos são versões

seccionadas do produto, mas eles realmente possuem capacidade para atender ao usuário e oferecem uma plataforma para avaliação do usuário”.

Com isso, cada entrega focará em uma ou mais funcionalidades, definidas através dos requisitos levantados inicialmente ou posteriormente. As fases de implementação do modelo incremental são possíveis de serem vistas na Figura 2.

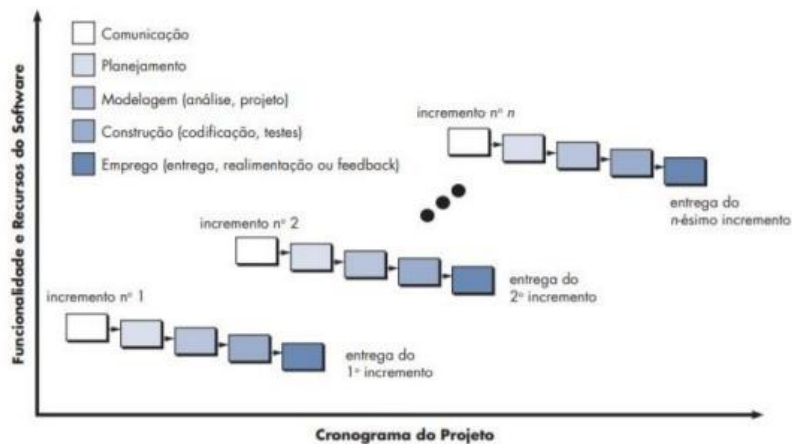


Figura 2: Representação do Modelo Incremental de Desenvolvimento.⁵

5. Desenvolvimento

Nessa seção serão apresentados os aspectos principais do desenvolvimento do sistema *FretExpresso*.

Inicia-se com a apresentação dos Casos de Uso, seguido do Fluxo Principal de Utilização, Banco de Dados Usuário/Motorista e finalizando com Protótipos de Tela.

5.1 Casos de Uso

USUÁRIO: Para acesso no sistema, é necessário que o usuário se cadastre na plataforma ou acesse através de um endereço já cadastrado. Após logado, ele verá um mapa exibido na tela, onde será possível ele especificar um ponto de origem e destino e buscar.

Com essa busca, será adicionado ao painel lateral informações relevantes sobre a viagem, como: ponto de origem e destino, o valor dessa viagem, a distância em

⁵ Diagrama retirado de PRESSMAN, Roger S. **Engenharia de software: uma abordagem profissional**

quilômetros e a duração da viagem. Nesse painel suspenso de viagens consultadas, é possível o usuário ordenar por menor duração ou por distância.

Após calculado as informações da viagem, o usuário tem a opção de solicitar a viagem, e, por praticidade, é possível o usuário cadastrar o seu endereço no sistema para que aquele sempre seja o seu endereço de origem de uma viagem.

A figura 3 mostra o Diagrama de Caso de Uso do Cliente referente à utilização do sistema e a figura 4 referente a utilização do motorista.

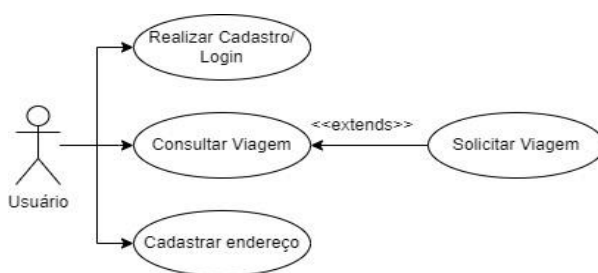


Figura 3: Caso de uso – Cliente

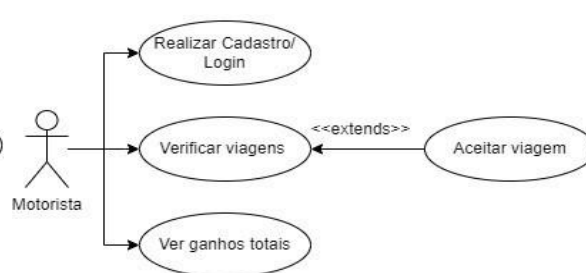


Figura 4: Caso de uso – Motorista.

MOTORISTA: Assim como os usuários, é necessário que o motorista tenha o cadastro na plataforma para acessar a página principal, onde poderá ver as viagens que foram solicitadas e aceitar ou não uma corrida. Além disso, será possível ter um histórico de todo o período na plataforma, onde será mostrado os ganhos totais obtidos durante o uso da plataforma.

5.2 Requisitos funcionais e não funcionais

Os requisitos funcionais, conforme exibido nos casos de uso são: Realizar cadastro/login; consultar viagens e solicitar viagens; Cadastrar endereço; Verificar viagens e aceita-las ou não; Ver ganhos totais.

Como requisitos não funcionais podemos citar: Segurança – Obtida a partir da utilização do Firebase Authentication, garante que apenas usuários cadastrados possam acessar a ferramenta; Responsividade – Obtida a partir da utilização do Vue.js, que consegue adaptar o layout de acordo com o tamanho do dispositivo acessado. Esse exemplo de reatividade pode ser notado na figura 5.



Figura 5: Reatividade fornecida pela utilização do Vue.Js

5.3 Fluxo Principal de Utilização

O usuário após logado na ferramenta é capaz de inserir a origem e o destino do frete, utilizando-se das funções da *API Places* do *Google Maps* de autocompletar endereço e de sugerir endereços próximos a sua localização. Com essas informações, é realizada uma chamada *HTTP* à *API Distance Matrix* que retorna qual a duração e distância dessa viagem.

Com essas informações, a ferramenta é capaz de calcular o valor dessa viagem de acordo com a distância dos pontos e salvar essa consulta em um banco de dados não relacional do *Cloud Firestore*.

Caso o usuário solicite a viagem, essa consulta é replicada para o banco de dados do motorista, que é capaz de aceitar ou não essa viagem. O diagrama dessa utilização está exemplificado na Figura 6 a seguir.

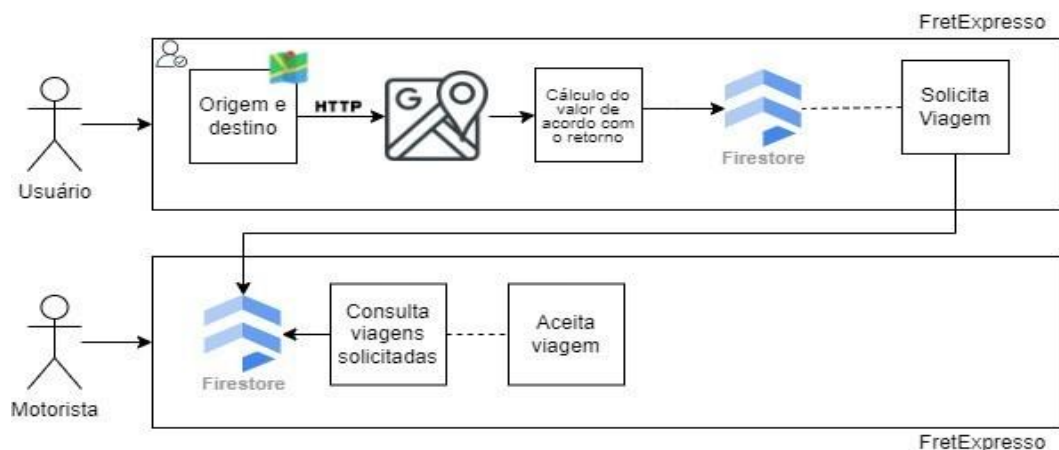


Figura 6: Diagrama de fluxo de utilização do FretExpresso.

5.4 Banco de Dados Usuário/Motorista

Como dito anteriormente, o banco de dados utilizado será o não relacional fornecido pela *Firebase*, o *Cloud Firestore*. Se trata de um banco de dados flexível e escalonável que armazena os dados em documentos, organizados em coleções.

Esse tipo de organização permite que os documentos suportem tipos de dados diferentes, que podem ir de simples textos à objetos aninhados.

Para o projeto, serão criadas três coleções: *routes* que será utilizada para armazenar informações das consultas realizadas pelos usuários, onde irá constar o valor da viagem, informações de destino e origem (endereço e coordenadas geográficas), a distância em quilômetros, a duração em horas/minutos e em segundos. A cada nova consulta, será adicionado um novo documento dentro dessa coleção, a fim de salvar uma consulta como um objeto único. Os itens inseridos na coleção são acessáveis via console do *Cloud Firestore* ou através da *API*, através da função *onSnapshot*.

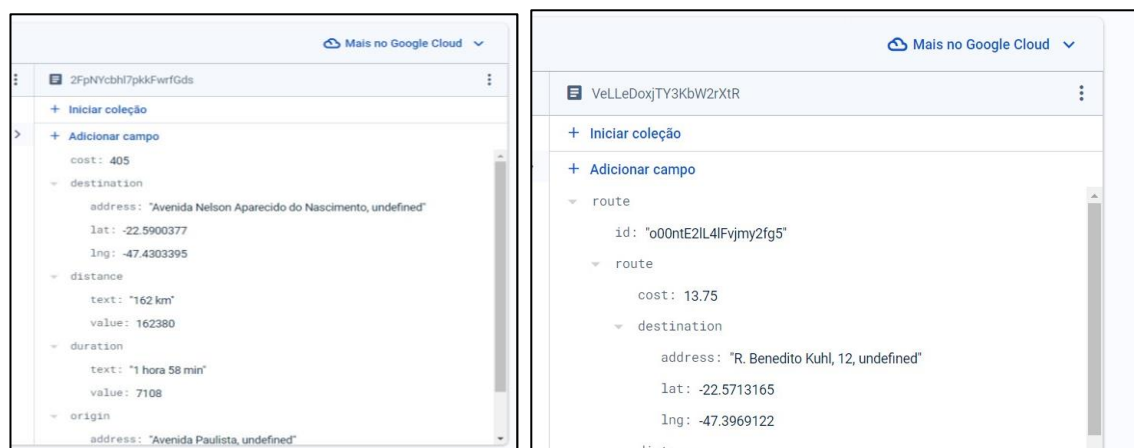


Figura 7: Coleção no FireBase: *routes* (rotas) e *viagensaceitas*.

A outra coleção a ser criada é a *viagensolicitadas*, que terá uma cópia do objeto da coleção *routes*, de acordo com a solicitada pelo usuário. Essa coleção será utilizada pelo motorista para identificar viagens que estão disponíveis para serem aceitas.

A última coleção a ser criada é a *viagensaceitas*, que são as viagens que de fato foram aceitas pelo motorista e que serão contabilizadas nos ganhos totais da plataforma.

A próxima subseção apresenta as interfaces gráficas (telas) desenvolvidas para o sistema FretExpresso.

5.5 Protótipos de Tela

A Figura 8 apresenta a Tela Inicial do Sistema FretExpresso.

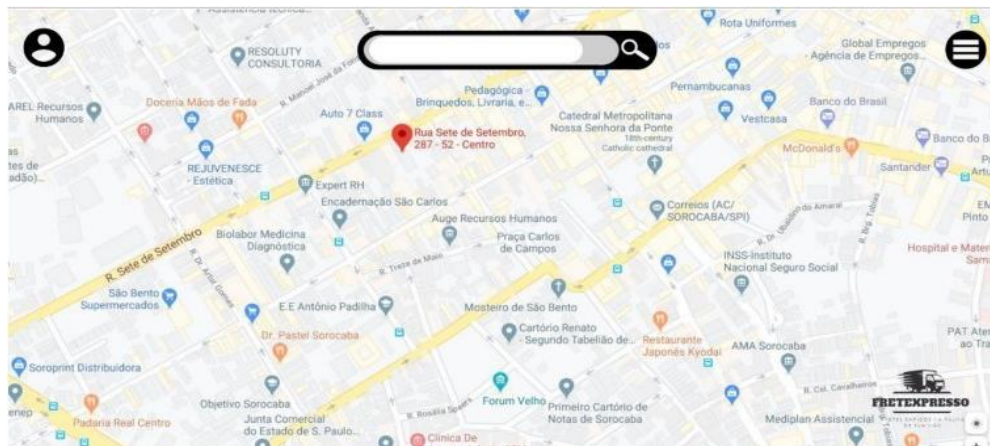


Figura 8: Protótipo da tela inicial do cliente.

A Figura 9 por sua vez, representa a tela de configurações para o Cliente.

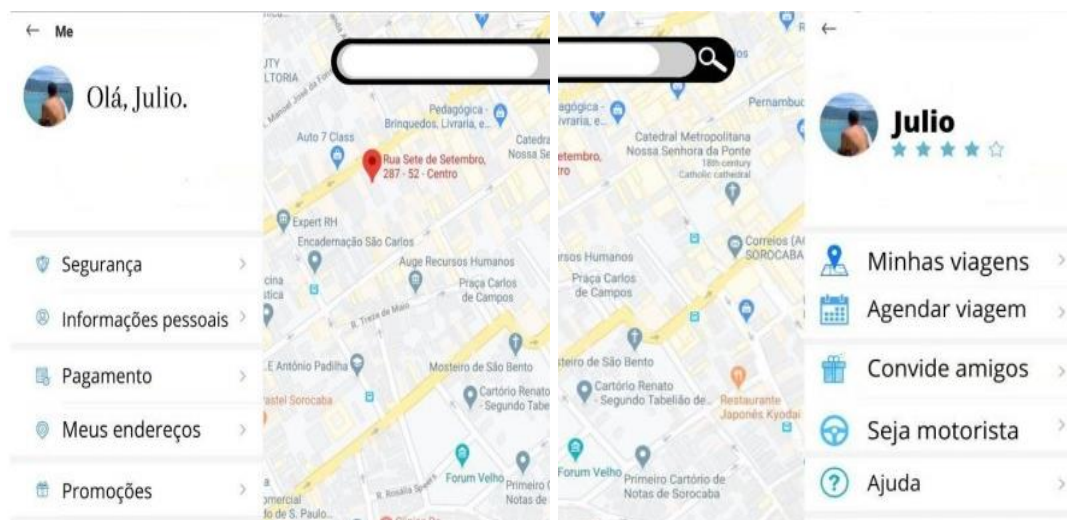


Figura 9: Protótipo da tela de configurações do cliente (a) e informações de viagens (b)

Na Figura 8, é possível observar um protótipo da tela inicial do sistema, onde mostra o mapa ao fundo e pode escolher o endereço pra onde o frete será realizado, além de poder acessar os ícones presentes nos cantos superiores, sendo o esquerdo acessar informações da conta como segurança e informações pessoais (Figura 9 - a) e no canto superior direito algumas informações relativas à viagem (Figura 9 - b).

6. Resultados

Com o desenvolvimento da pesquisa foi possível implementar o sistema *web* de maneira local, sendo possível, tanto o usuário quanto o motorista, realizarem os casos de usos de acordo com o proposto. A seguir representam-se as interfaces gráficas apresentadas aos usuários do sistema dos tipos “**Cliente**” e “**Motorista**”.

6.1 Jornada Cliente

Na página inicial do Cliente é possível a realização de um *login* com um usuário cadastrado no *Cloud Firebase Authentication* e a opção de cadastro (caso não possua) e a opção de *login* motorista, caso o perfil seja esse, conforme ilustrado na Figura 10.

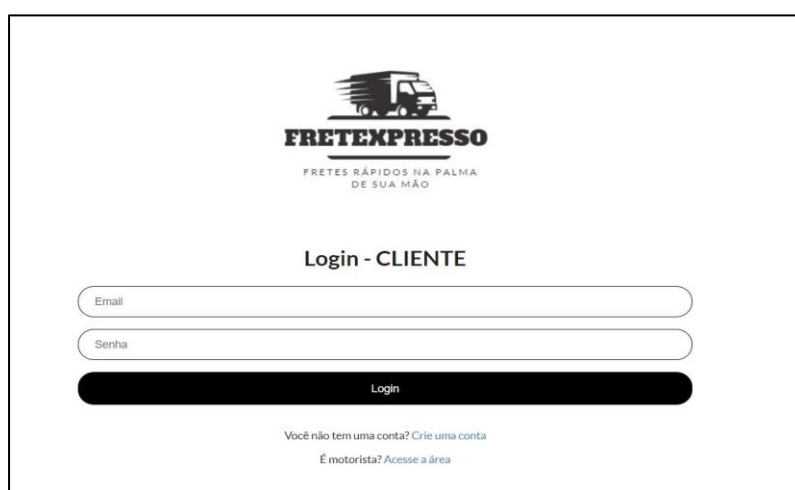


Figura 10: Login Cliente.

Caso o cadastro não seja encontrado no *Cloud Firebase Authentication*, é disparada uma mensagem de erro informando que o usuário não existe, conforme representado na Figura 11.



Figura 11: Erro ao realizar login cliente.

Caso o *login* seja realizado com sucesso, é apresentada ao cliente a tela principal (Figura 12).

Nessa tela, é aberto um mapa, centralizado na Rua Thereza Rosolen Mazzarat, 28 Jardim Boa Vista, Limeira – SP. Nessa tela estão presentes dois campos, onde é possível o usuário inserir os pontos de origem e destino de sua viagem. Em cada um desses campos está ativada a opção de *Autocomplete* do Google Maps, onde as alternativas sugeridas estejam próximas às coordenadas geográficas definidas.

A demonstração da tela inicial pode ser observada na figura 12 abaixo.

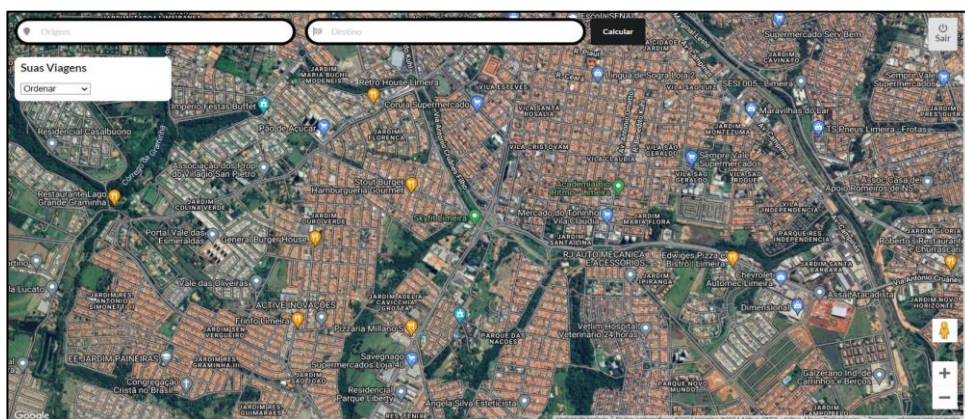


Figura 12: Tela Inicial do Cliente do sistema **FretExpresso**.

Ao apertar em **Calcular**, a viagem é adicionada a “Suas Viagens” (na coleção *routes*) já com o cálculo do preço, que é feito a partir da distância entre os pontos, sendo cada km R\$2,50 ($const \text{ valor} = \text{distancia} * 2.5$, mostrado na figura 14).

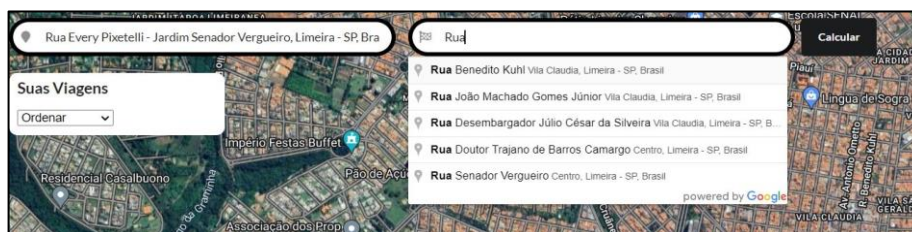


Figura 13: Exemplo de *Autocomplete* e botão **Calcular**.

```
//Tratativa da distancia para realização do cálculo do custo
const dist = elements[0].distance.text.split(" ")[0];
const distancia = dist.replace(/,/g, '.');

//cálculo do valor de acordo com a distância
const valor = distancia * 2.5;
this.route.cost = valor;
```

Figura 14: Exemplificação no código da implementação do cálculo do custo

Além disso, temos apresentada à esquerda a lista de viagens já calculadas pelo usuário anteriormente, onde são apresentadas as informações: **origem**, **destino**, **valor da viagem**, **distância de um ponto ao outro (em Kms)**, a **duração da viagem** e o **Status** “Não Solicitada” até que seja solicitada de fato pelo usuário.

Caso o usuário clique em **Solicitar**, o status é alterado para “Solicitada” e o registro é enviado à coleção **viagensolicitadas** e o botão de Solicitar é desativado.

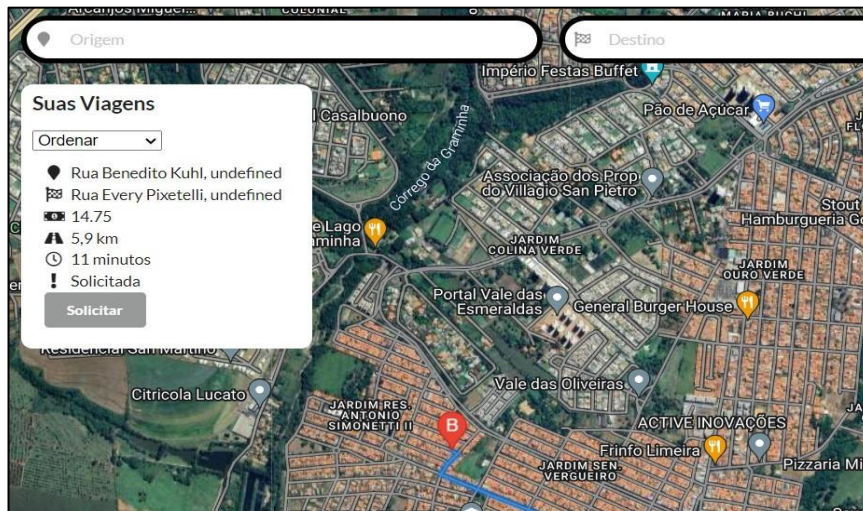


Figura 15: Tela do FretExpresso com a opção “Suas Viagens” acionada.

Ao clicar em uma viagem em específico, ela é exibida no mapa, evidenciando qual a rota traçada, conforme representado na Figura 16.

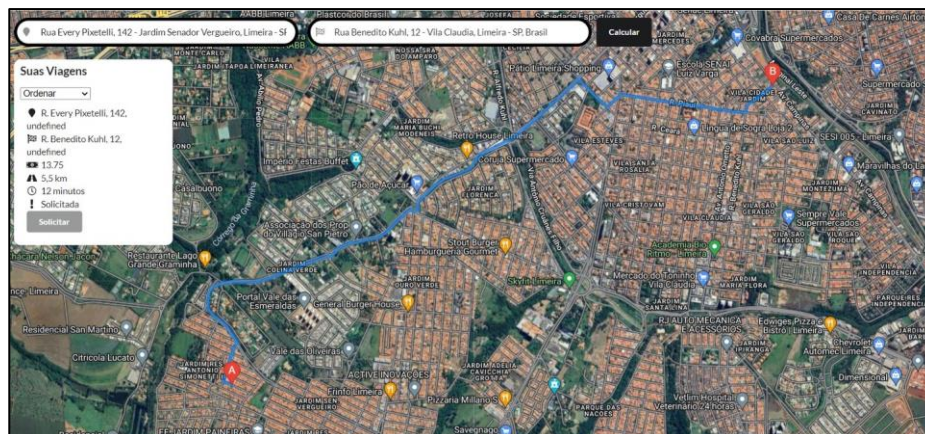


Figura 16: Exemplo de viagem traçada.

Temos também nessa tela o botão sair, apresentado no canto superior direito, que realiza o *logout* do usuário e redireciona para a página de *login* do cliente, representada na Figura 10.

As próximas telas são relativas à interação do usuário do tipo **Motorista** com o sistema **FretExpresso**.

6.2 Jornada Motorista

A Figura 17 representa a tela inicial de login para o usuário do tipo **Motorista**.



A imagem mostra a interface de login para o usuário do tipo Motorista. No topo, há o logotipo da FretExpresso, que inclui um ícone de caminhão e o slogan "FRETEXPRESSO FRETES RÁPIDOS NA PALMA DE SUA MÃO". Abaixo do logotipo, o título "Login - MOTORISTA" é exibido. O formulário de login contém dois campos de entrada: "Email" e "Senha". Abaixo dos campos, há um botão de "Login" em um fundo preto com o texto "Login" em branco. Na base do formulário, há duas opções de link: "Você não tem uma conta? Crie uma conta" e "É usuário? Acesse a área".

Figura 17: Login Motorista.

Assim como o no caso do Cliente, há uma página dedicada para o *login* do motorista, a qual possui conexão com o *Cloud Firebase Authentication*. Nessa tela apresenta-se a opção de cadastro (caso o motorista não possua) e a opção de *login* Cliente, caso o perfil seja esse. Caso o cadastro não seja encontrado no *Cloud Firebase Authentication*, é disparada uma mensagem de erro, assim como no login do cliente, informando que o usuário não existe, exemplificado na Figura 11.

Caso o *login* seja realizado com sucesso, apresenta-se ao usuário a **Tela Principal do Motorista** (Figura 18), onde nessa tela são disponibilizados quatro botões principais intitulados: “**Viagens Solicitadas**”, “**Viagens Aceitas**”, “**Ganhos Totais**” e “**Sair**”.

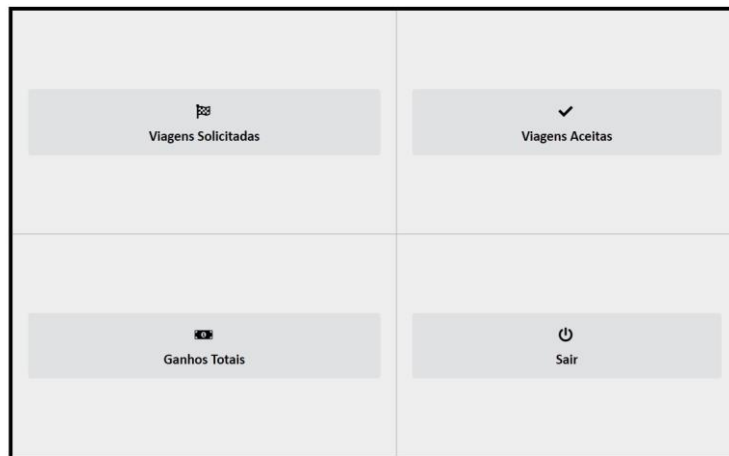


Figura 18: Tela Inicial do Motorista.

Nessa tela “**Viagens solicitadas**” são mostradas as viagens que foram solicitadas pelos clientes e que ainda não foram aceitas por um motorista. Ao exibir a lista, é exibido a opção **Aceitar** ou **Recusar**.

Quando uma viagem é **aceita** por um motorista, ela é copiada para a coleção **viagensaceitas** e o status é atualizado para “Aceita por um motorista” e mostrado ao cliente (Figura 19). Caso a viagem seja **recusada** o registro é apagado da coleção **viagenssolicitadas** e não é mostrado mais na tela Viagens Solicitadas do motorista.

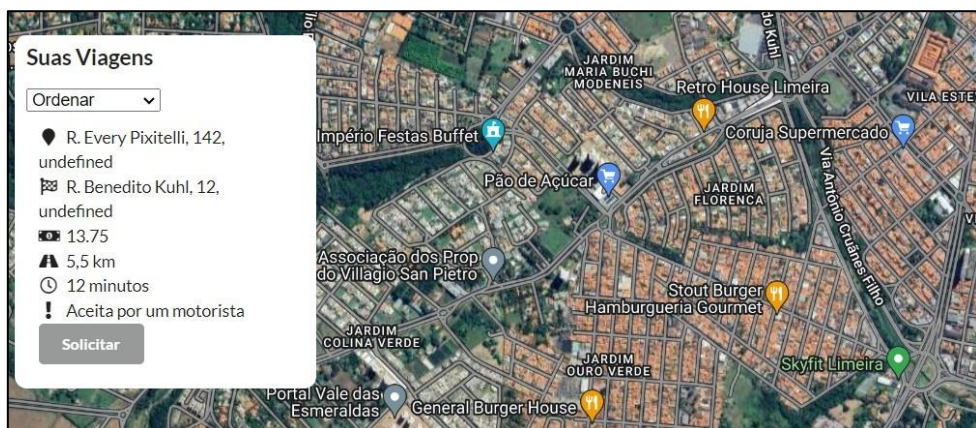


Figura 19: Atualização do status após uma viagem ser aceita.

Ao se clicar em “**Viagens Aceitas**”, serão mostradas as viagens que foram aceitas pelo motorista. A Figura 20 representa a tela de “Viagens Aceitas” (a) e a de “Ganhos Totais” (b).

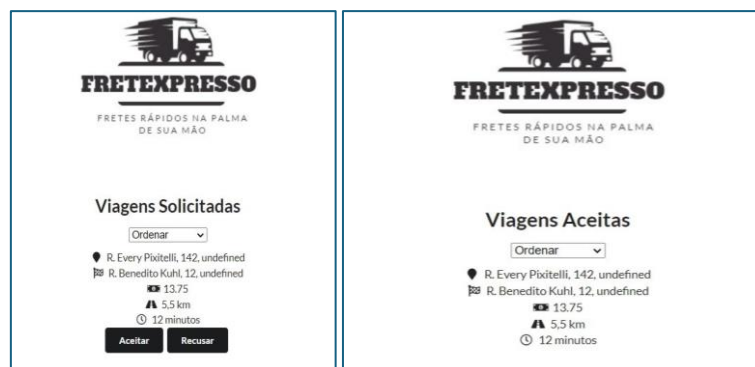


Figura 20: Representação de telas. Viagens Solicitadas (a) e Viagens Aceitas (b)

A funcionalidade “**Ganhos Totais**” apresenta ao motorista quantas viagens ele aceitou, bem como a soma de seus valores, contabilizando o total ganho dentro da plataforma, conforme representado na Figura 21.



Figura 21: Representação tela “Ganhos Totais”

Por fim, a funcionalidade “**Sair**” permite ao motorista realizar o *logout* da ferramenta e redireciona para a página de login, representada na Figura 17.

7. Conclusões

O presente trabalho teve como objetivo desenvolver um *software* que permitisse a solicitação de frete de maneira mais eficiente e de uma forma mais centralizada, além da maneira clássica de divulgação em redes sociais. A proposta foi baseada na premissa de que a solicitação de frete por meio de redes sociais, faz com que o solicitante precise buscar anúncios referentes a prestadores que realizam esse tipo de trabalho e verificar se há disponibilidade no dia.

Com o desenvolvimento do *FretExpresso*, essas solicitações podem acontecer de maneira mais fluida, onde a solicitação de frete pode ser aceita pelo motorista que

tenha disponibilidade aquele dia, já informando o valor aproximado da corrida. Além disso, o sistema pode trazer vantagens também para o motorista, que tem uma visão geral dos valores obtidos de fretes através da ferramenta e a quantidade de viagens realizadas.

O presente trabalho oferece uma solução alternativa e centralizada para esse tipo de solicitação de serviço. Além disso, o trabalho possibilitou a utilização e experimentação de um *framework* em ascensão, o *Vue.js*, utilizado para criação de telas estáticas com reutilização de componentes, trazendo mais rapidez durante o seu desenvolvimento. As *APIs* do *Google Maps* também possuem uma ampla documentação sobre a sua utilização e facilidade na implementação em um projeto.

Em termos do trabalho de conclusão de curso como um todo, ele permitiu ao aluno o emprego de conhecimentos adquiridos em diversas disciplinas do Curso Superior em Análise e Desenvolvimento de Sistemas, tais como: Linguagem de Programação, Banco de Dados, Engenharia de Software e Desenvolvimento *Web*, Metodologia de Pesquisa Científica, Projeto de Sistemas 1 e Projeto de Sistemas 2.

Como trabalhos futuros, podemos citar a reformulação da parte visual do *software* para chegar ao mais próximo do apresentado nos protótipos, a inclusão de funcionalidades como: Configurações, cadastro de cartões, acompanhamento de viagem em andamento e implementação de *chat* entre cliente e motorista.

8. Referências Bibliográficas

AMAZON. **O que é SDK? - Explicação sobre SDK - AWS**. Disponível em: <<https://aws.amazon.com/pt/what-is/sdk/>>. Acesso em: 16 mar. 2024.

CAMARGO, D. A. D. O. **Estudo das técnicas de desenvolvimento WEB e validação deste estudo com um portal para recicladores e Produtores de reciclados**. Disponível em: <https://repositorio.pgsskroton.com.br/bitstream/123456789/1116/1/artigo%2032.pdf>. Acesso em: 25 ago. 2020.

CONTROLOG. **Controlog – Gestão de Entregas em Tempo Real**. Disponível em: <https://www.appcontrolog.com.br/> Acesso em: 13 de março de 2024.

DE SOUZA, I. **Afinal, o que é JSON e para que ele serve? Descubra agora!** Disponível em: <<https://rockcontent.com/br/blog/json/>>. Acesso em: 16 mar. 2024.

DOUTBOX. **Conheça as vantagens de utilizar o framework Vue.js.** Setembro, 2023. Disponível em: <https://doutbox.com.br/blog/conheca-as-vantagens-de-utilizar-oframework-vuejs#:~:text=Reatividade%20e%20Interatividade:%20Uma%20das,necessidade%20de%20recarregar%20a%20página>. Acesso em: 13 jan. 2024.

ERLE, Schuyler; GIBSON, Rich. **Google Maps Hacks.** São Paulo: O'Reilly Editora, 2006.

FIREBASE. **Firestore Realtime Database.** Disponível em: <https://firebase.google.com>. Acesso em 13/03/2024.

FIREBASE. **Cloud Firestore.** Disponível em: <https://firebase.google.com/docs/firestore?hl=pt-br>. Acesso em: 13 jan. 2024.

FIREBASE. **Authentication.** Disponível em: <https://firebase.google.com/docs/auth?hl=pt-br>. Acesso em: 13 Jan. 2024.

FLANAGAN, David. **JavaScript : o guia definitivo.** David Flanagan ; tradução: João Eduardo Nóbrega Tortello ; revisão técnica: Luciana Nedel. – 6. ed. – Porto Alegre : Bookman, 2013. xviii, 1062 p. : il. ; 25 cm.

GOOGLE. **Documentação da Plataforma Google Maps.** Disponível em: <https://developers.google.com/maps/documentation?hl=pt-br>. Acesso em: 13 jan. 2024.

IBGE – Instituto Brasileiro de Geografia e Estatística. **Pesquisa Nacional por Amostra de Domicílios. 2020.**

PRESSMAN, Roger S. **Engenharia de software: uma abordagem profissional /** Roger S. Pressman ; tradução: Ariovaldo Griesi ; revisão técnica Reginaldo Arakaki, Julio Arakaki, Renato Manzan de Andrade. – 7. ed. – Dados eletrônicos. – Porto Alegre : AMGH, 2011.

UBER Blog. **Envie e receba itens com Uber Flash.** Disponível em: <https://www.uber.com/pt-BR/blog/uber-flash-solicite-viagens-para-enviar-artigospeessoais-por-meio-do-app-da-uber/>. Acesso em: 13 jan. 2024.

Documento Digitalizado Público

Artigo final de TCC do aluno Júlio César Leite de Jesus

Assunto: Artigo final de TCC do aluno Júlio César Leite de Jesus
Assinado por: Leandro Ledel
Tipo do Documento: Relatório
Situação: Finalizado
Nível de Acesso: Público
Tipo do Conferência: Cópia Simples

Documento assinado eletronicamente por:

- Leandro Camara Ledel, PROFESSOR ENS BASICO TECN TECNOLOGICO, em 18/03/2024 15:55:17.

Este documento foi armazenado no SUAP em 18/03/2024. Para comprovar sua integridade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifsp.edu.br/verificar-documento-externo/> e forneça os dados abaixo:

Código Verificador: 1615520

Código de Autenticação: 4105f6164d

