

Turis Info

Aline Rezendes Giroto¹, Gabriel Paniz Patzer (Orientador)¹

¹Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul -
Câmpus Sertão - Rodovia RS 135, Km 32,5 - Distrito Eng. Luiz Englert - CEP:
99170-000 - Sertão-RS

23205@aluno.sertao.ifrs.edu.br, gabriel.patzer@sertao.ifrs.edu.br

Abstract. *The article describes the development of a software that assists in the management of charters, trips and passengers. The project is built around the concepts of web responsiveness and serverless, using the React Native library and the Firebase platform, with the objective of improving the control of trips, vehicles, contracts and passengers, aiming at a better organization and growth in business management. This project was developed as the Term Paper of the Análise e Desenvolvimento de Sistemas course from Instituto Federal de Educação Ciência e Tecnologia do Rio Grande do Sul - Câmpus Sertão.*

Resumo. *O artigo descreve o desenvolvimento de um software que auxilia no gerenciamento de fretamentos, viagens e passageiros. O projeto é construído voltado aos conceitos de web responsividade e serverless, utilizando a biblioteca React Native e a plataforma Firebase, tendo como objetivo aprimorar o controle das viagens, veículos, contratos e passageiros visando uma melhor organização e crescimento na gestão do negócio. Esse projeto foi elaborado como Trabalho de Conclusão do Curso de Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação Ciência e Tecnologia do Rio Grande do Sul - Câmpus Sertão.*

1. Introdução

Através deste artigo é apresentado o Trabalho de Conclusão de Curso (TCC) de Análise e Desenvolvimento de Sistemas (ADS), do Instituto Federal de Educação Ciência e Tecnologia do Rio Grande do Sul - Câmpus Sertão. Neste, a temática abordada é o desenvolvimento de um *Software* nomeado de Turis Info, com o objetivo de gerenciar viagens, mensalidades, contratos e estudantes de uma empresa de fretamento. Uma vez que se nota um crescente movimento de alunos se deslocando a outras cidades em busca de estudo e formação complementar, há um déficit de softwares que contemplem todos os pontos levantados no projeto, como a relação dos horários de ida e volta dos passageiros diariamente.

Ao longo do projeto, serão utilizados os conceitos de responsividade, que conforme Marcotte (2010) significa, “desenhar para o fluxo e refluxo das coisas”, ou seja, desenvolver para uma experiência de visualização ideal, que incorpora tecnologias que renderiza mídia adaptáveis e não só flexíveis. Também, será empregado no desenvolvimento do sistema, a metodologia *Serverless* que, de acordo com Sousa (2020), seria uma evolução da computação em nuvem, consistente no escalonamento dos recursos, sendo responsabilidade do provedor de serviço. Além do mais, as taxas por recursos ociosos das funções não são tarifadas, diminuindo os valores em geral, fazendo com que se torne mais acessível em relação ao custo benefício.

Esse trabalho é subdividido em seções, das quais a seção 2 retrata qual o

problema com sua definição e justificativa. Já na seção 3, é abordado o objetivo principal e os específicos que este trabalho vai contemplar. A seção 4, mostra quais softwares estão disponíveis no mercado com um intuito semelhante ao proposto no projeto, analisando pontos positivos e limitações de cada um. Na seção 5, será tratada a metodologia utilizada para a construção do projeto e as soluções encontradas para a problemática. Em seguida, na seção 6, são apresentados os resultados e, por fim, na seção 7, serão citadas as considerações finais sobre o trabalho e trabalhos futuros.

2. Definição do problema e justificativa

Na sociedade atual, buscar novos horizontes e oportunidades fora de seu domicílio tornou-se algo comum. Tanto quanto buscar meios de locomoção para atingir novos níveis de escolarização, muitas vezes fora do âmbito em que se convive, esse deslocamento também é conhecida como “movimento pendular” que, segundo Melo e Vala (2004) do Instituto Nacional de Estatística (INE), seria uma expressão para designar o movimento diário da população entre o local de residência para o trabalho ou estudo. Com isso, constata-se um aumento na procura de transporte para esses fins e, para exemplificar essa relação, um levantamento publicado pelo aplicativo ClickBus registrou um aumento de 116% nas vendas online de passagens rodoviárias no ano de 2022 [Guarulhos Hoje 2022].

Com esse aumento significativo pela busca de transporte, algumas empresas privadas começaram a prestar o serviço de fretamento, com horários e turnos específicos de viagens a outras cidades e instituições para transportar alunos e trabalhadores. Segundo Vasconcellos (2012), o fretamento pode ser dividido em três classes: o contínuo, referente ao transporte regular de empregados; o eventual, relativo aos deslocamentos eventuais, normalmente associados a eventos e festas; e o escolar, referente ao transporte de alunos de forma regular, porém com características e formas de operação distintas das demais. Sendo assim, essas empresas necessitam de software de apoio na gestão do negócio, pois conforme Alves (2013) aponta, a importância do uso desses sistemas apresenta uma consequência na melhoria dos processos de administração empresarial bem como evita investimentos inadequados para a organização.

Essa classe de *software* é do tipo aplicação que, de acordo com Pressman (2011), um *software* de aplicação é um programa que auxilia na solução de uma necessidade específica de negócio. Consequentemente, uma vez que esses *softwares* fazem uso de bases de informações e controlam as funções de negócio, acabam por contribuir com a tomada de decisões das empresas, facilitando as operações comerciais. Portanto, qualquer empreendimento requer um sistema adequado ao seu negócio, consequentemente as companhias de fretamento de veículos, tendem a necessitar de apoio de *softwares* para a gestão de suas relações. Porém, muitos desses sistemas não atendem todos os requisitos solicitados pela empresa contratada, fazendo com que alguns utilizem meios de gerenciamento manual ou com sistemas que não oferecem todo apoio e integração na gestão do serviço.

Desta forma, a proposta desse projeto é a criação de um sistema *web* que auxilia o gerenciamento de uma empresa de transporte rodoviário de estudantes e trabalhadores, que incorpore um espaço de administração para controle de clientes, veículos, mensalidades, viagens e contratos. Também será disponibilizado um espaço do qual possibilite aos clientes a consulta de suas mensalidades e o registro de seus horários, para que a empresa possa obter um controle diário de viagens por turno, juntamente com uma interface que estimule e satisfaça o usuário na utilização do mesmo, possibilitando ainda o acesso a diversos dispositivos. Para isso, o trabalho proposto será baseado na ideia de *design web* responsivo, que, segundo Marcotte (2010), é o projeto de uma experiência de visualização ideal, incorporando tecnologias baseadas em padrões para torná-los não apenas mais flexíveis, como adaptáveis à mídia que os renderiza, além de mais acessíveis e com maior usabilidade.

3. Objetivos

O objetivo geral do projeto é desenvolver um *software web* responsivo e compatível com os diversos dispositivos, para isso é utilizada a metodologia *serverless*. Deve auxiliar no gerenciamento de transporte de passageiros, que incorpore um controle de contratos, veículos, viagens e mensalidades. Além disso, contempla os seguintes objetivos específicos:

- Disponibilizar ao usuário uma interface de fácil compreensão e uso.
- Permitir o controle de horários das viagens diárias.
- Controlar as informações dos veículos, bem como a disponibilidade para o transporte, considerando o número de passageiros.
- Permitir o gerenciamento das mensalidades e pagamentos pelos passageiros e administradores.

4. Trabalhos relacionados

Ao analisar os programas disponíveis no mercado para o auxílio do gerenciamento da gestão do transporte, seja de alunos e trabalhadores, veículos e viagens, não foi encontrado nenhum sistema que atendesse todos os requisitos necessários. Foram analisados alguns sistemas, como o “Sistema Eletrônico de Gestão do Transporte Escolar” (SETE), um *software* desenvolvido pelo Centro Colaborador de Apoio ao Transporte Escolar (CECATE) junto à Universidade Federal de Goiás (UFG), em parceria com o Fundo Nacional de Desenvolvimento da Educação (FNDE) e a e-governança. O *software* é voltado a auxiliar na gestão do transporte escolar brasileiro, considerando as singularidades de cada região [Engenharia de Transporte 2021], bem como a possibilidade de construir rotas. Esse sistema possui alguns objetivos semelhantes aos que são propostos no projeto, como o controle de clientes, frota e relatórios (Figura 1). Porém, não foi possível identificar um controle de horários e mensalidades, nem por parte da administração nem dos passageiros.



Figura 1. Tela inicial do software SETE.

Outra referência de *software* é o “Tra-Fretamento”, desenvolvido pela empresa Soriodev, para a gestão de empresas de fretamento [Soriodev 2022]. Esse programa possui uma versão disponível de forma gratuita para que os clientes façam testes e experimentem o produto. Porém, esta versão impõe limitações nas quantidades de cadastros e interações, além de cobrar uma licença e taxas para atualização. O sistema conta com o gerenciamento de clientes, fornecedores, controle de frota, fretamento e estoque, mas não possui a administração de mensalidades e viagens com horários (Figura 2). Também não possui um espaço para o passageiro consultar seus dados e mensalidades.

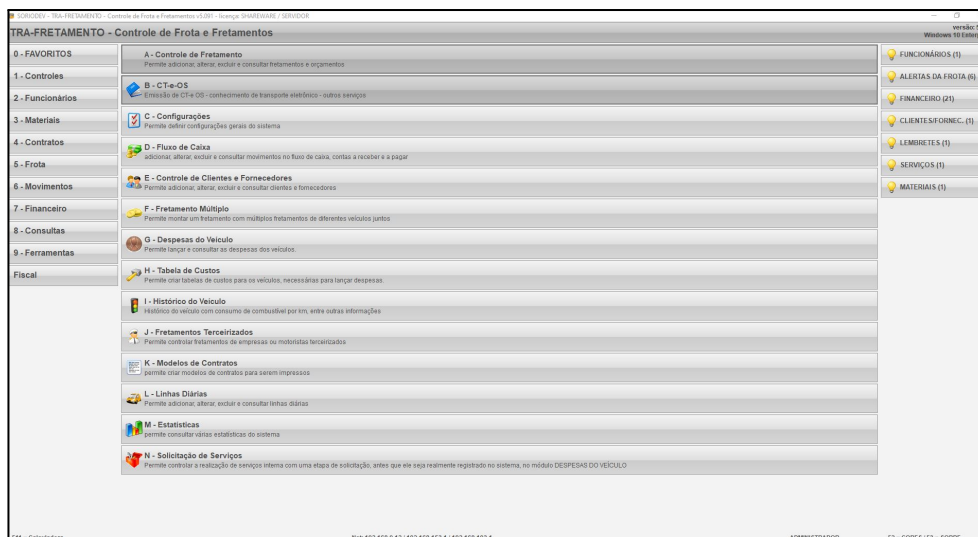


Figura 2. Tela inicial do software Tra-Fretamento.

Além do mais, não possui uma interface agradável ao usuário (Figura 3), conforme um ponto de vista de Interação humano-computador (IHC), que se dedica a estudar os fenômenos de interação entre pessoas e sistemas computacionais seja eles quais forem [SBC 2021]. Por esse motivo, faz com que possivelmente venha a

manutenção dos dados, através do *CRUD* (acrônimo de *Create, Read, Update and Delete*), sendo as operações básicas de um sistema para gerenciar as informações pertinentes, sejam elas de passageiros, de veículos, das instituições, dos contratos, das viagens e das mensalidades. Além disso, conta com acesso a relatórios diários emitidos, de acordo com informações inseridas dos usuários, com os horários das viagens e o total dos passageiros de cada turno. Também, é possível visualizar os relatórios pertinentes a clientes cadastrados, mensalidades e suas especificações e “*status*” de pagamento.

5.2 Definição de requisitos

O levantamento de requisitos é fundamental para o processo de desenvolvimento de software. Conforme Sommerville (2011), requisitos de um sistema são as descrições do que o programa realiza, suas operações e restrições de funcionamento, podendo ser divididos em funcionais e não funcionais. Machado (2016) aborda que os requisitos expressam as restrições e as características do *software* para satisfazer as expectativas dos usuários, independente da tecnologia empregada para o desenvolvimento.

Para melhor detalhar os requisitos analisados, a seguir apresenta-se os requisitos funcionais e não funcionais.

5.2.1 Requisitos funcionais

De acordo com Sommerville (2011), requisitos funcionais são as especificações dos serviços que o sistema deve fornecer ou como se comportar em determinada situação, podendo contemplar aspectos mais gerais ou específicos, dependendo do caso analisado. Com isso, os requisitos funcionais descritos neste projeto para melhor compreendê-lo são:

- **Manter usuários:** A partir do momento em que o usuário é registrado no sistema, ele passa a usufruir das funcionalidades do software. Para isso, terá um sistema de autenticação através de um login e senha, para controlar os acessos e aumentar a segurança das operações.
- **Manter mensalidades:** Quando um passageiro assina o contrato passa a ter mensalidades a pagar devido à utilização do serviço da empresa. Com isso, fazer o *CRUD* desse serviço é essencial, para isso, será preenchido um formulário onde é relacionado o passageiro e seus dados com as informações pertinentes ao contrato e pagamentos.
- **Manter passageiros:** Para auxiliar a administração com os dados de quem utiliza o transporte, no caso os passageiros, será realizado todo controle de cadastros, manutenções e exclusão de informações de cada indivíduo. Um formulário com os dados dos passageiros como: nome completo, CPF, RG, estado civil, endereço, instituição de ensino, anexar uma foto ou arquivo do contrato assinado, valor da mensalidade calculada, dias e turnos que irá utilizar o transporte. Com essas informações, será possível fazer a administração dos passageiros e relacionar com outras funcionalidades do sistema, como por exemplo as viagens, os contratos e mensalidades.

- **Manter veículos:** A administração contará com um espaço para gestão de veículos da empresa e, conseqüentemente, melhoria na tomada de decisões em relação a cada viagem e cada veículo utilizado. Com isso, terá o registro das informações como: marca/modelo, consumo por litro, situação do veículo e regularização da documentação, capacidade e data de fabricação.
- **Manter contratos:** Os contratos são estabelecidos no início de cada semestre ou ano letivo, são importantes pois declaram os direitos e deveres do contratante e do contratado caso necessário posteriormente.
- **Manter instituições:** Com as informações das instituições que a empresa atende, ou seja, até qual local oferece o transporte, é possível calcular o valor das mensalidades e destino das rotas para a melhor organização da empresa. Com isso, será cadastrado o nome e endereço dos locais das instituições atendidas.
- **Gerar relatório de viagens:** Será emitido um relatório diário com a relação entre as viagens e os passageiros, com o registro separado por turno e horários de ida e volta, para auxiliar na tomada de decisões na escolha do melhor veículo para o transporte e também na ajuda com os possíveis enganos de passageiros ou retornos.
- **Gerar relatório de passageiros:** O relatório de passageiros é uma relação da quantidade de clientes ativos que a empresa possui, isso ajuda a identificar o público alvo e a relação com as instituições atendidas pelo transporte.
- **Gerar relatório de mensalidades:** Sobre o relatório de mensalidades, é uma apresentação sobre as mensalidades ativas, ajudando no controle das situações financeiras da empresa. Com isso, a empresa pode ter uma relação com a situação financeira da companhia, emitir histórico e alertas de pagamento aos passageiros.
- **Linkar o *Whatsapp* para retirada de dúvidas dos passageiros:** Para os passageiros tirarem as dúvidas sobre qualquer assunto relacionado ao transporte, terá disponível na sessão do passageiro um “botão” que direciona ao *whatsapp* (aplicativo de troca de mensagens instantâneas *online*) de um responsável e auxiliar da empresa para iniciar um diálogo.
- **Efetuar login:** Realizar login é essencial em um sistema *web*, tanto para o controle de acesso quanto para a segurança do sistema e as informações veiculadas. Para isso, qualquer usuário deverá realizar o login, através de um nome e senha já cadastrados e únicos, para terem acesso a qualquer funcionalidade e serviço do sistema.
- **Gerar pagamentos:** O passageiro terá apoio para efetuar o pagamento de suas mensalidades, para isso, o sistema vai dispor do pagamento através do pix, tecnologia recente que possibilita o pagamento direto a qualquer momento, e também o boleto bancário gerado pela empresa e enviado particularmente.
- **Consultar mensalidades:** Os passageiros podem consultar a qualquer momento

as suas situações financeiras pendentes ou não com a empresa. Com isso, será implementado um relatório para a consulta das mensalidades, tanto pagas quanto pendentes, para auxiliar os passageiros a gerenciar seus pagamentos.

- **Registrar viagens:** Os passageiros poderão registrar suas viagens diariamente, com o dia, nome e horários de ida e volta. Esse registro irá auxiliar na construção do relatório de viagens aos administradores e na tomada de decisões.

5.2.2 Requisitos não funcionais

Sommerville (2011) argumenta que os requisitos não funcionais são as restrições aos serviços ou funções presentes no *software*. Não são requisitos relacionados diretamente aos serviços propriamente ofertados pelo sistema, no entanto, requisitos e funções que podem delimitar a implementação do programa. Com esse levantamento, os requisitos não funcionais apresentados no projeto são:

- Para auxiliar na construção do projeto, evitar problemas de versionamento e armazenamento do código desenvolvido, foi utilizado a plataforma do GitHub. Ela auxilia a ter o controle de entregas, alterações e hospedagem, de forma gratuita e remota.
- A fim de cumprir com o objetivo de um sistema *web* responsivo, foram utilizadas as bibliotecas *React Native* e *ReactJs* baseadas em *JavaScript*, para a construção do projeto. Uma vez que facilitam a atingir o objetivo principal, a portabilidade entre dispositivos. Também foi utilizado a ferramenta *Expo*, que facilita na construção de aplicativos, uma vez que permite o acesso à API nativa do dispositivo sem precisar instalar qualquer dependência, provendo o código base para rodar a aplicação do sistema operacional.
- Aplicar o paradigma *serverless* no desenvolvimento do projeto, o benefício dessa metodologia é a flexibilidade de acesso ao serviço, ou seja, pode comunicar com diversas plataformas e dispositivos. Além de não precisar se preocupar com o escalonamento e cobrança por tempo ocioso. Para isso, foram utilizadas algumas ferramentas oferecidas pela plataforma do *Firebase*, que até certo limite se torna gratuita, prevendo a implementação de *back-end* sem gerenciamento de servidor e melhorar a qualidade de acesso aos aplicativos [Firebase 2022], também podendo hospedar o próprio site pela plataforma.
- Com o objetivo de manter o paradigma de *serverless*, decidiu-se pelo banco de dados *Cloud Firestore* do *Firebase*, plataforma provida pela *Google*, uma vez que esta garante ser um banco de dados flexível, NoSql, escalável para armazenamento e sincronização de dados em nuvem [Firestore 2022].
- Utilização da ferramenta *Figma*, recurso que facilita a modelagem e criação das telas e formatos visuais do sistema. Proporciona a exportação como código para o aproveitamento e auxílio na codificação do sistema e tomada de decisões.

5.3 Diagrama de Casos de Uso

O modelo de Casos de Uso, segundo Stadzisz (2002), é um instrumento de eficiência para a documentação e construção dos serviços de um sistema, também auxilia no entendimento e comunicação com os clientes na definição dos requisitos. Com isso, o diagrama de casos de uso é representado com o uso de atores (usuários do sistema), serviços ou funcionalidades do sistema e os relacionamentos entre eles. Também apresenta uma linguagem simples e de fácil compreensão para atender todos os usuários [Guedes, 2011].

A seguir, apresenta-se o diagrama de casos de uso (Figura 4) referente ao projeto proposto. Logo após, mostra-se o detalhamento de alguns casos de uso do sistema. O detalhamento de alguns casos de uso é realizado no Apêndice A.

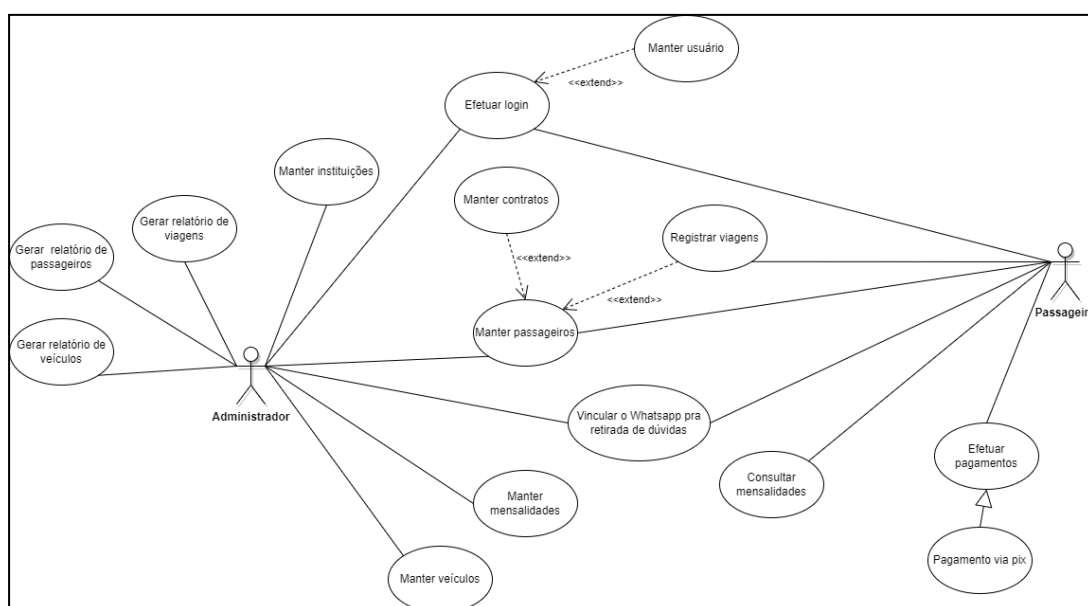


Figura 4. Diagrama de casos de uso do projeto.

5.3.1 Detalhamento dos casos de uso

Para melhor compreender o diagrama de caso de uso, a seguir serão apresentadas as Tabelas 1, 2 e 3, detalhando os casos de uso e suas características, começando pelo caso de uso “Efetuar Pagamentos”.

Tabela 1. Detalhamento do caso de uso “Efetuar pagamentos”

Nome	Efetuar pagamentos
Atores	Passageiro
Descrição	O passageiro opta por realizar pagamento da mensalidade através de duas opções. Depois de escolher e efetuar o pagamento, será atualizado “ <i>status</i> ” da mensalidade no sistema.

Pré-condições	O passageiro deve estar registrado, logado no sistema e a mensalidade tem que estar disponível para pagamento.
Pós-condições	O passageiro seleciona a forma de pagamento, o Pix “copia e cola” ou escaneando o <i>QR Code</i> .

Em seguida, é exposto o detalhamento do caso de uso “Registrar Viagens” pertencente a uma funcionalidade do sistema.

Tabela 2. Detalhamento do caso de uso “Registrar viagens”

Nome	Registrar viagens
Atores	Passageiro
Descrição	O passageiro registra no sistema a data e horários de ida e volta das viagens, diariamente de acordo com os dias que irá utilizar o transporte.
Pré-condições	O usuário deve estar cadastrado e logado no sistema.
Pós-condições	Após registrar os dados, o sistema envia o relatório diário de viagens para o administrador ter uma relação da quantidade de lugares nos veículos.

Em seguida, é apontado o detalhamento do caso de uso nomeado como “Manter Passageiros”, uma funcionalidade muito importante ao funcionamento do sistema em geral.

Tabela 3. Detalhamento do caso de uso “Manter passageiros”

Nome	Manter passageiros
Atores	Passageiro e administrador
Descrição	Manutenção de passageiros e seus respectivos dados no sistema.
Pré-condições	O passageiro deve ter assinado o contrato com a empresa.
Pós-condições	Após o cadastro, o passageiro passa a usufruir das funcionalidades como interação com o registro de viagens, consulta de mensalidades e as opções de pagamento das mesmas.

5.4 Diagrama de Classes

O modelo de Diagrama de Classes, de acordo com Guedes (2011), abrange a definição das estruturas das classes que o sistema utiliza, estabelecendo métodos, atributos e seus relacionamentos. Fowler (2003) determina também, que o diagrama mostra as propriedades, operações e as restrições que se aplicam aos objetos conectados.

Dessa forma, apresenta-se o diagrama de classes (Figura 5) construído em concordância com o projeto, no qual, pode-se analisar as classes, objetos e relações dos sistema.

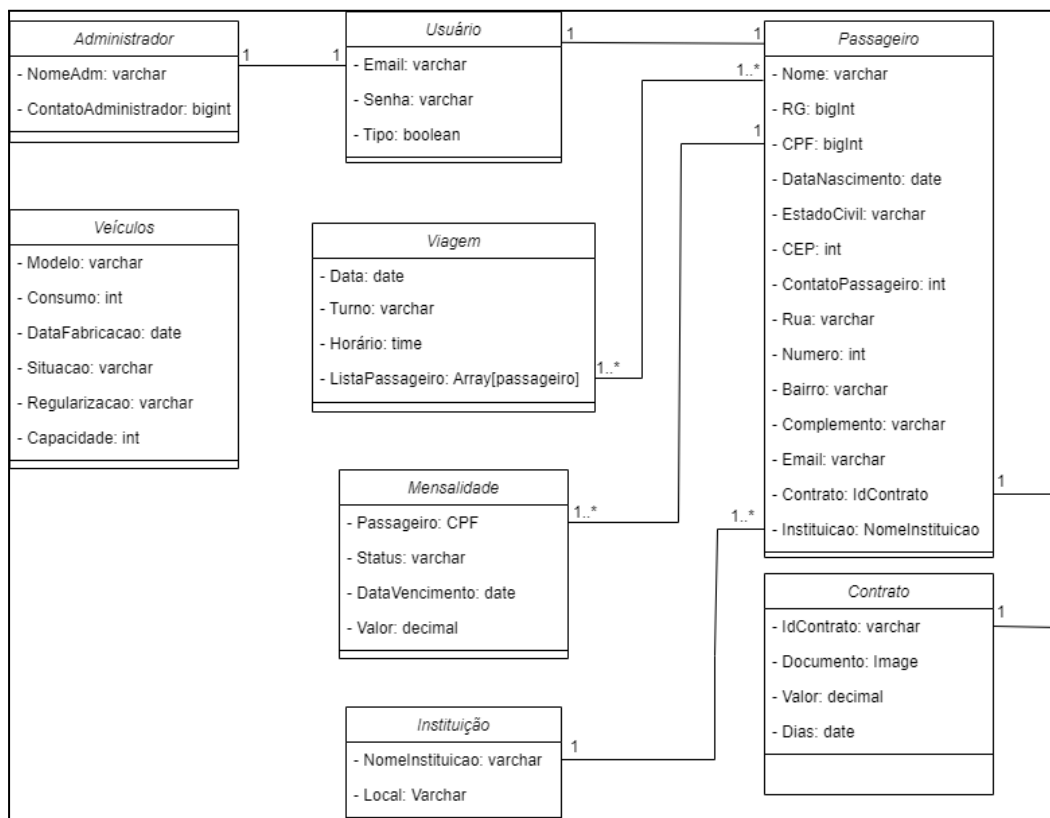


Figura 5. Diagrama de Classes do projeto.

5.5 Representação da interface

Um projeto de interface é importante no desenvolvimento do projeto, uma vez que estipula com qual ambiente o usuário irá ter contato e como ele se portará dentro do sistema, sempre visando uma boa usabilidade do software. Como Pressman (2011) alega, se a interface for bem projetada o usuário flui através das interações sem grande esforço. Caso contrário, o resultado é a frustração e baixa eficiência do projeto. Dessa forma, foi elaborado um projeto de interface através da ferramenta *Figma*. Essa representação gráfica foi construída em uma disciplina anterior a codificação do projeto, por essa razão, a representação se difere do resultado aplicado.

Na imagem apresentada (Figura 6), modelou-se a interface do sistema *Turis Info* com algumas funcionalidades e interações, tanto em relação ao administrador quanto ao passageiro. Para diferenciar as telas modeladas, foram utilizadas cores para distinguir web, com a borda vermelha, para o mobile, com a borda verde. Porém, anteriormente era visado em um sistema que seria implementado sem a utilização do *React Native*, por isso há uma diferença de componentes e interfaces que serão comparadas posteriormente, mas as funcionalidades são as mesmas.

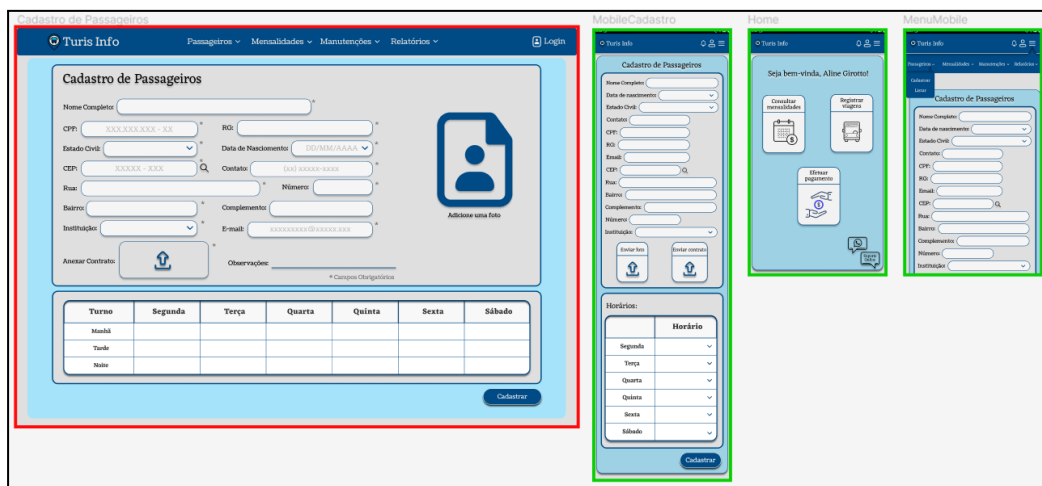


Figura 6. Modelagem de telas do sistema.

A primeira interface planejada (Figura 6, borda vermelha), é relacionada ao administrador, que possui o cadastro das informações pertencentes aos passageiros que desejam adquirir o serviço. Desse modo, contam com um formulário para a inserção de dados importantes para manter o passageiro vinculado à empresa, facilitando a organização dos horários de utilização do transporte e a relação com demais funcionalidades no sistema. Na interface *mobile* (Figura 6, borda verde) existe a mesma funcionalidade anterior, cadastro de passageiros, mas redirecionada ao dispositivo móvel. Podemos analisar como seria a disposição dos itens e comportamento em tela menor.

Na Figura 7, tela de cadastro de passageiros, podemos ter a representação do sistema implementado, onde notamos algumas diferenças de design gráfico, porém as funcionalidades se mantêm ao que foi proposto anteriormente.

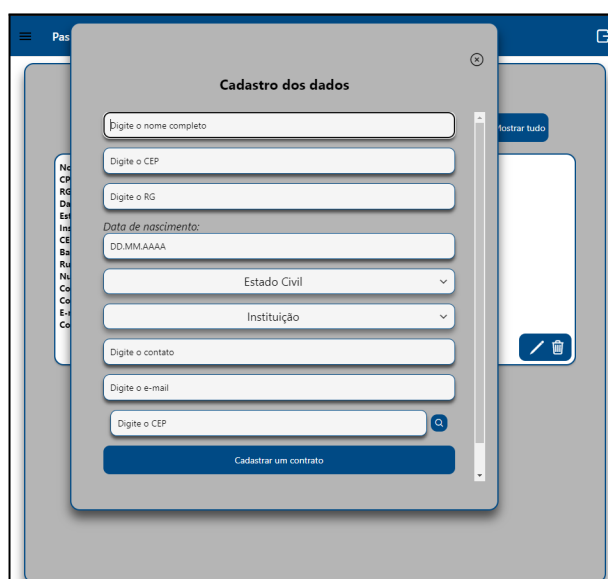


Figura 7. Tela de cadastro de passageiros.

6 Resultados

Para atingir a proposta, foi utilizado o Firebase como uma ferramenta de auxílio e complementação do sistema, como a autenticação e armazenamento dos dados, ainda sendo um banco de dados que armazena através de documentos não relacionais, como representado na imagem (Figura 8), facilitando a manipulação dos mesmos. Também foi escolhido pois oferece funcionalidades gratuitas, até certa quantidade de leituras e edições, para a implementação do projeto.

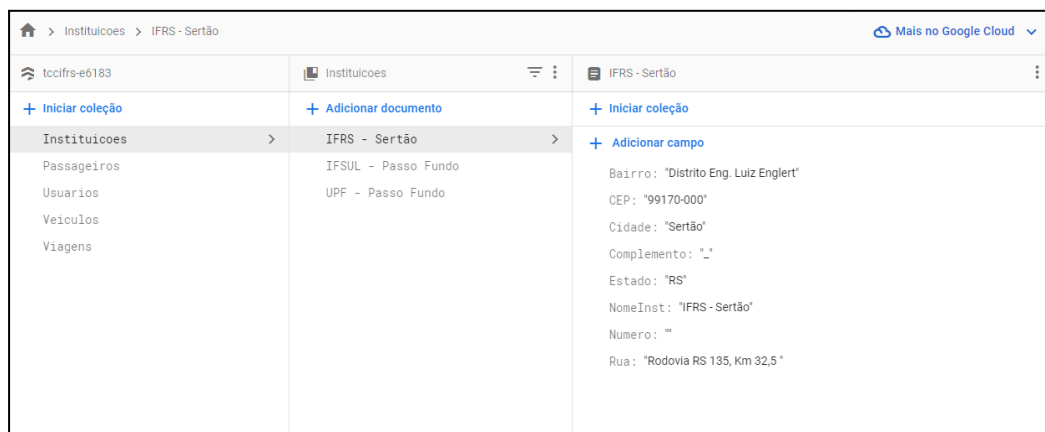


Figura 8. Exemplo de organização do Cloud Firestore.

Outra plataforma utilizada foi a LivePix, como plataforma de auxílio e intermédio de gerenciamento dos pagamentos providos do pix. Uma vez que a plataforma, mesmo sendo voltada para *live streams*, provê de um *QR Code* e uma chave “copia e cola” da conta, também oferece uma dashboard com os dados de cada pagamento, usuários e possibilita uma transferência posterior à conta pessoal do usuário [LivePix 2022].

Com base nas ferramentas auxiliares e bibliotecas e linguagens de programação citadas, os resultados alcançados e métodos utilizados contribuíram para o desenvolvimento de um sistema web que auxilia o gerenciamento do negócio e soluciona a problemática analisada, tanto para administradores quanto para passageiros.

Na Figura 9 podemos visualizar uma das interfaces do administrador, nela está implementado o requisito de manter veículos, onde observamos a listagem, com os dados pertencentes, opção de exclusão e edição, os novos cadastros são por meio de um modal (Figura 10) para a inserção dos dados a serem acrescentados ou modificados. As demais funcionalidades pertencentes aos gestores, como manter passageiros, mensalidades e instituições, e se dispõem da mesma forma que a da imagem, alterando-se apenas as respectivas informações. Também, ao analisar a diferença entre os usuários e o sistema, foi implementado o módulo de gestão mais voltado ao *web* e *Desktop*, enquanto o módulo do passageiro é mais voltado ao uso *mobile*, seja em sistema Android ou iOS, porém ambos atuam da mesma forma em todas as plataformas.

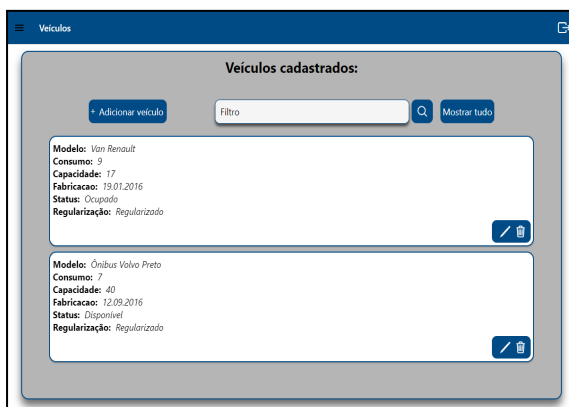


Figura 9. Tela de veículos no sistema.

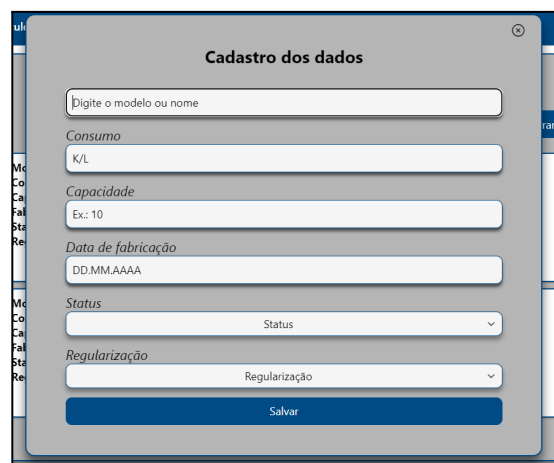


Figura 10. Modal de cadastro de veículos.

As representações a seguir estão implementadas no módulo do passageiro. De suas funcionalidades, podemos analisar na figura da esquerda (Figura 11) o requisito de registro de viagens, com o cadastro das informações pertinentes. Ao lado (Figura 12), podemos analisar a interface inicial do sistema para passageiros, que conta com as funcionalidades a eles dedicadas.

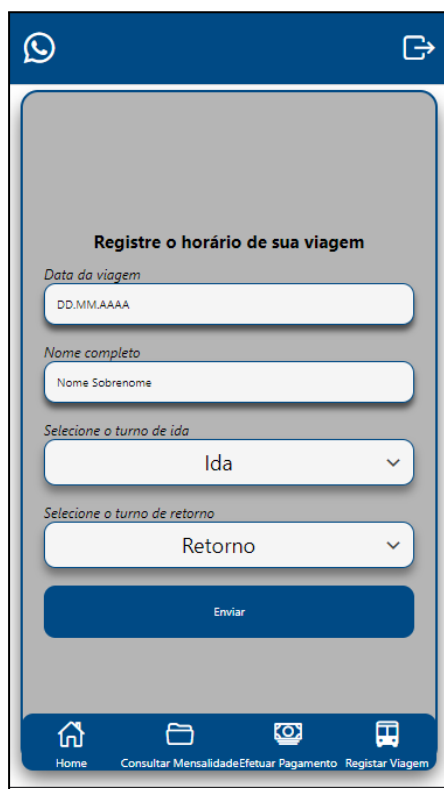


Figura 11. Tela de registro de viagem.

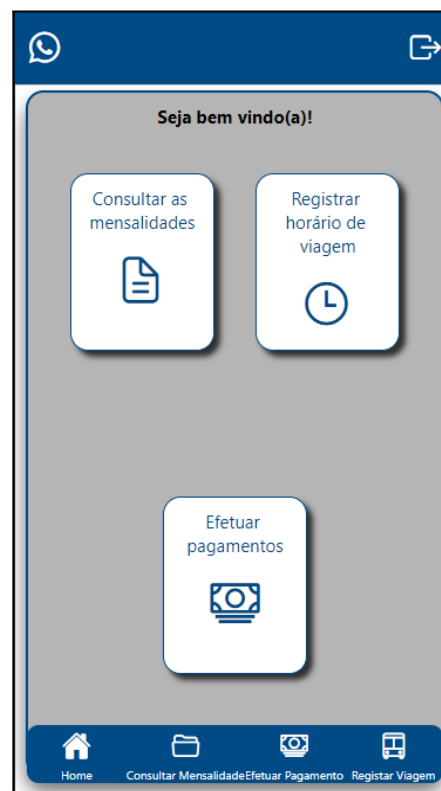


Figura 12. Tela inicial do módulo de passageiro.

7 Considerações finais

Com o desenvolvimento do projeto deste TCC, nomeado Turis Info, foi possível colocar em prática os conhecimentos adquiridos ao longo do curso, lembrando alguns assuntos estudados, como requisitos, construção de diagramas e utilização de algumas ferramentas já debatidas, bem como o *Figma* na modelagem das interfaces. O curso ofereceu uma ótima base de ensino para a construção do projeto, visando tanto a engenharia de software quanto a programação.

Os maiores desafios encontrados foram:

- Relembrar e construir os diagramas apresentados, pois são tratados em disciplinas anteriores ao projeto final, muitas vezes não tão específicas e complexas como o realizado.
- A procura por softwares semelhantes ao projeto, uma vez que é notado a falta de sistemas para esse mercado com as funcionalidades em específico.
- Aprender a utilizar o *Firebase* e suas funcionalidades, no qual não havia utilizado anteriormente na ementa do curso, mas é notável ser de grande auxílio e base para o projeto.
- A codificação utilizando o *React Native*, biblioteca de programação ensinada mais ao final do curso.
- Devido ao projeto inicialmente ser modelado com um intuito e metodologia diferente, sem utilização do *Firebase* e a biblioteca *React Native*. Para se manter nos objetivos propostos, foi refeita toda a análise e modelagem para atender às finalidades, tornando um pouco trabalhoso recomeçar o planejamento com tão pouco tempo.

O projeto cumpre com o objetivo inicial, um sistema *web* que facilita a administração do negócio, controlando as viagens, os passageiros, veículos e mensalidades. Não só, mas facilitando os registros e manipulação dos horários registrados das viagens diárias, tanto por administradores quanto por passageiros. Da mesma forma que atinge a proposta de sistema desenvolvido visando o paradigma *serverless* e responsividade, atendendo a todos dispositivos e facilidade de acesso a qualquer momento e local.

Contudo, espera-se que o projeto atenda de forma satisfatória aos objetivos propostos, facilitando o gerenciamento e comunicação tanto entre sistema e usuários quanto a administradores e passageiros.

7.1 Trabalhos futuros

Para trabalhos futuros, sugere-se acrescentar ao sistema funcionalidades pertencentes a viagens de turismo. Tanto ao administrador, para controlar essas informações, quanto ao usuário solicitar o serviço à empresa informando quais são os informes da viagem solicitada.

Uma outra sugestão seria uma compatibilidade com um sistema externo de leitura

de cartão magnético, uma vez integrado com o banco de dados, pode oferecer um melhor controle da quantidade de passageiros a bordo dos veículos e monitoramento de mensalidades atrasadas. Também, acrescentar uma forma de pagamento através da própria plataforma, diversificando os métodos de pagamento do serviço utilizado. Uma sugestão, seria acrescentar pagamentos com cartão de crédito e débito ou até uma forma de implementar um saldo em conta, facilitando os débitos das faturas de mensalidade.

Além disso, integrar a *API (Application Programming Interface)* da plataforma LivePix ao sistema, possibilitando a automação na confirmação dos pagamentos recebidos pela plataforma. Por fim, fazer melhorias na interface gráfica do sistema, a tornar mais agradável ao usuário.

Referências

Alves, C. A. L. (2013) “A Importância Da Tecnologia Da Informação Nas Empresas”, Revista Científica Semana Acadêmica, 24ª edição, <https://semanaacademica.org.br/artigo/importancia-da-tecnologia-da-informacao-nas-empresas>.

Engenharia de Transportes, FCT e UFG (2021) “SETE Sistema Eletrônico de Gestão do Transporte Escolar”, <https://transportes.fct.ufg.br/p/31448-sete-sistema-eletronico-de-gestao-do-transporte-escolar>.

Firebase (2022) “Aprenda os fundamentos”, <https://firebase.google.com/docs>.

Firestore (2022) “Cloud Firestore”, <https://firebase.google.com/docs/firestore?hl=pt>, Novembro.

Fowler, M. (2003) “UML Distilled: A Brief Guide to the Standard Object Modeling Language”, Addison-Wesley Professional, 3ª edição.

Guarulhos Hoje (2022) “Busca por passagens rodoviárias apresenta crescimento de 116% em comparação com 2021”, <https://www.guarulhoshoje.com.br/2022/01/26/busca-por-passagens-rodoviaras-apresenta-crescimento-de-116-em-comparacao-com-2021/>, Janeiro.

Guedes, G. T. A. (2011) “UML2: uma abordagem prática”, São Paulo: Novatec editora, 2ª edição.

LivePix (2022) “Home”, <https://livepix.gg/>.

Machado, F. N. R. (2016) “Análise e gestão de requisitos de software: onde nascem os sistemas”, 3ª edição.

Marcotte, E. (2010) “Responsive Web Design”, <https://alistapart.com/article/responsive-web-design/#>, Maio.

Melo, C. e Vala, F. (2004) “Movimentos Pendulares e Organização do Território

Metropolitano: Distâncias e Proximidades nos Sistemas Metropolitanos de Lisboa e Porto”, Revista Portuguesa de Estudos Regionais, 5ª edição, <https://review-rper.com/index.php/rper/article/view/121>.

Pacífico, R. D. G., Duarte, L. F. S., Castanho, M. S., Nacif, J. A. M. e Vieira, M. A. M. (2020) “Sistema de processamento de pacotes Serverless”, <https://sol.sbc.org.br/index.php/sbrc/article/view/12282> , Dezembro.

Pressman, R. S. (2011) “Engenharia de software: uma abordagem profissional”, Porto Alegre: Bookman, 7ª edição.

SBC (2021) “Interação Humano-Computador”, <https://www.sbc.org.br/14-comissoes/390-interacao-humano-computador>, Novembro.

Sommerville, I. (2011) “Engenharia de software”, São Paulo: Addison Wesley, 9ª edição.

Soriodev (2022) “Software para gestão de fretamentos”, <https://soriodev.com.br/software-gestao-de-fretamentos/>.

Sousa, F. R. C. (2020) “Computação Serverless e Gerenciamento de Dados”, Simpósio Brasileiro de Banco de Dados (SBBD), <https://sol.sbc.org.br/index.php/sbbd/article/view/13641>.

Stadzisz, P. C. (2002) “Projeto de software usando a UML”, Centro Federal de Educação Tecnológica do Paraná, Departamento acadêmico de informática, <http://www.etelg.com.br/paginaete/downloads/informatica/apostila2uml.pdf>.

Vasconcellos, E. A. (2012) “Transporte por fretamento: os serviços de transporte de passageiros por fretamento”, Caderno técnico, volume 9, p.26-45, http://files-server.antp.org.br/_5dotSystem/download/dcmDocument/2013/03/11/2C71B08A-6A12-4356-AA3F-F5A0F0DFB717.pdf.