

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE CIÊNCIA DA COMPUTAÇÃO

AUGUSTO BORANGA

**UFRGS Caronas: um aplicativo móvel de
caronas para a comunidade da UFRGS**

Monografia apresentada como requisito parcial
para a obtenção do grau de Bacharel em Ciência
da Computação

Orientador: Profa. Dra. Renata Galante

Porto Alegre
2020

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Rui Vicente Oppermann

Vice-Reitora: Prof^a. Jane Fraga Tutikian

Pró-Reitor de Graduação: Prof. Vladimir Pinheiro do Nascimento

Diretora do Instituto de Informática: Prof^a. Carla Maria Dal Sasso Freitas

Coordenador do Curso de Ciência de Computação: Prof. Sérgio Luis Cechin

Bibliotecária-chefe do Instituto de Informática: Beatriz Regina Bastos Haro

AGRADECIMENTOS

Primeiramente, agradeço à minha família. Em especial meus avós, minha mãe e meu padrasto, por sempre estarem lá para mim. Não teria conseguido sem vocês. Também às minhas bisas Elfrida e Helena, que sempre me incentivaram e ajudaram nos estudos. Se cheguei aqui hoje, devo muito a vocês todos.

Agradeço à minha orientadora Renata Galante pela ajuda na realização deste trabalho e por ser essa pessoa tão querida e atenciosa. Também ao pessoal do CPD, que me acolheu muito bem e me ajudou a crescer muito, pessoal e profissionalmente. Especialmente, agradeço à minha chefe e co-orientadora Bárbara Arena, por todos os ensinamentos e conselhos, e ao colega Abel Corrêa, pela ajuda com as alterações na API de dados.

Não posso deixar de agradecer também aos meus amigos, que me ajudaram a segurar as pontas nas horas mais difíceis. Em especial, Christian Schmitz, Isadora Oliveira, Lucas Flores, Lúcio Franco e Matheus Pereira. Vocês são fodas.

Por fim, agradeço ao meu irmão Antônio Devos, essa pessoinha incrível que me ajudou a ver de novo a luz nos dias mais escuros.

Muito obrigado.

RESUMO

Em Porto Alegre, todos os dias, milhares de pessoas se locomovem pela cidade em direção a algum campus da UFRGS para estudar ou trabalhar. Nos horários de pico, como início da manhã e fim da tarde, as ruas ficam congestionadas devido à grande quantidade de ônibus e carros nelas transitando. Pessoas que utilizam carro para realizar o trajeto poderiam compartilhar eventual espaço sobrando em seu carro para dar carona a pessoas que fazem um trajeto parecido. Isso ajudaria a desafogar o trânsito, traria mais conforto no deslocamento das pessoas que ganharam carona e promoveria em certo nível a integração da comunidade da UFRGS. O objetivo deste trabalho é fornecer uma ferramenta que faça o intermédio entre pessoas da comunidade da UFRGS que estejam dispostas a dar caronas em seu trajeto e outras que gostariam de receber caronas para se deslocar até algum dos campi, ou partir deles. Para isto, foi definido como objetivo o desenvolvimento de um aplicativo móvel que posteriormente pode vir a ser disponibilizado para a comunidade da UFRGS. A avaliação de usabilidade do aplicativo foi feita através da aplicação de testes com usuários. Os resultados mostram que o protótipo desenvolvido possui interface intuitiva, e as principais funcionalidades implementadas receberam boas avaliações.

Palavras-chave: Caronas. mobilidade. UFRGS. iOS.

Implementation of hitchhiking mobile app for UFRGS community

ABSTRACT

In Porto Alegre city, thousands of people commute towards some of UFRGS's campi every day either to study or work. At peak time, such as early morning and evening, the streets get crowded due to the large amount of buses and cars moving through them. People who use their car to make their journey to study or work could use some space left to give a ride to people who take a similar path. This would help reduce traffic, bring more comfort to the commuting of the hitchhikers, and promote the integration of the university community to some extent. The purpose of this work is to provide a tool for the UFRGS community, mediating those who are willing to give rides and the ones who would like to hitchhike to or from a UFRGS campus. To this end, the development of a mobile application to be made available later to the community was set as the objective. The product usability evaluation was made through user testing. The results show that the prototype app created has an intuitive interface, and the main functionalities got good ratings.

Keywords: UFRGS. Commute. Ride. iOS.

LISTA DE FIGURAS

Figura 2.1	Exemplo de fluxo de aplicativo criado utilizando o Interface Builder	17
Figura 2.2	Arquitetura Cliente-servidor presente no sistema do UFRGS Caronas	19
Figura 2.3	Arquitetura MVC presente no APP	20
Figura 2.4	Fluxo da comunicação entre uma aplicação e um banco de dados usando o padrão <i>Repository</i>	21
Figura 2.5	Etapas do desenvolvimento de <i>software</i> segundo o Modelo Cascata	23
Figura 4.1	Comunicação entre os elementos do projeto UFRGS Caronas	30
Figura 4.2	Modelagem original do UFRGS Caronas	31
Figura 5.1	Captura de tela de quadro do Trello	35
Figura 5.2	Modelagem atualizada do UFRGS Caronas	37
Figura 6.1	<i>Onboarding</i> do UFRGS Caronas	40
Figura 6.2	Tela de login	40
Figura 6.3	Tela do Modo Caronista	41
Figura 6.4	Tela com os detalhes da carona e o <i>status</i> do pedido, que neste caso é “Pendente”	42
Figura 6.5	Tela de filtros para a busca de caronas	42
Figura 6.6	Filtro booleano disponível apenas para usuárias mulheres	43
Figura 6.7	Tela de busca de caronas quando não há resultados para a busca	43
Figura 6.8	Tela mostrada quando a busca gerou resultados	44
Figura 6.9	Tela com detalhes da carona resultante da busca	45
Figura 6.10	Tela onde o usuário adiciona uma mensagem para o motorista	45
Figura 6.11	Tela que mostra pro usuário que o pedido da carona foi feito com sucesso	46
Figura 6.12	Tela que mostra a lista de oferecimentos de carona criados pelo usuário	46
Figura 6.13	Tela que mostra os detalhes da carona oferecida	47
Figura 6.14	Tela que mostra um oferecimento de carona com solicitações respondidas (aceitas e recusadas)	48
Figura 6.15	Tela com novas solicitações de carona	48
Figura 6.16	Tela onde o usuário responde a um pedido de caronas	49
Figura 6.17	Telas antes e depois da escolha da origem e do destino da carona	49
Figura 6.18	Tela onde o usuário informa detalhes do ponto de encontro da carona	50
Figura 6.19	Telas onde o usuário escolhe o local de origem e de destino da carona, respectivamente	50
Figura 6.20	Telas antes e depois da escolha da data e horário da carona	51
Figura 6.21	Janelas de seleção da data e horário da carona	52
Figura 6.22	Última tela de cadastro de carona, com escolha do carro, observações e número de vagas	52
Figura 6.23	Telas de escolha de carro para a carona e adição de observações, respectivamente	53
Figura 6.24	Opção disponível para usuárias mulheres, que quando selecionada indica que a carona sendo cadastrada deverá ser vista apenas por outras mulheres	53
Figura 6.25	Tela que indica sucesso ao cadastrar a carona	54
Figura 6.26	Perfil do usuário	55
Figura 6.27	Histórico de caronas oferecidas pelo usuário	55
Figura 6.28	Histórico de caronas solicitadas pelo usuário	56
Figura 6.29	Tela que mostra a lista de carros do usuário	56

Figura 6.30 Tela de edição de um carro do usuário	57
Figura 6.31 Tela de cadastro de um novo carro	57
Figura 7.1 Gráfico dos gêneros declarados pelos participantes da avaliação com usuários	61
Figura 7.2 Gráfico das idades dos participantes da avaliação com usuários.....	61
Figura 7.3 Gráfico do nível de escolaridade dos participantes da avaliação com usuários	62
Figura 7.4 Gráfico do nível de experiência dos participantes com o uso da internet.....	62
Figura 7.5 Gráfico do nível de experiência dos participantes com o uso de <i>smartphone</i>	63
Figura 7.6 Aplicativos utilizados pelos participantes.....	64
Figura 7.7 Avaliação da realização de <i>login</i>	64
Figura 7.8 Avaliação do cadastro de um carro	65
Figura 7.9 Avaliação do oferecimento de uma carona	66
Figura 7.10 Avaliação da busca por caronas com filtro	66
Figura 7.11 Avaliação da criação de um pedido de carona.....	67
Figura 7.12 Avaliação do oferecimento de carona apenas para mulheres	68
Figura 7.13 Avaliação da busca de caronas apenas para mulheres	68
Figura 7.14 Avaliação da tarefa de excluir um oferecimento de carona	69
Figura 7.15 Avaliação da tarefa de fazer <i>logout</i> do aplicativo	70

LISTA DE TABELAS

Tabela 3.1 Plataformas em que a aplicação está disponível.....	27
Tabela 3.2 Funcionalidades disponíveis.....	27
Tabela 3.3 Opções de segurança	28

LISTA DE ABREVIATURAS E SIGLAS

API Application Program Interface

MVC Model-View-Controller

DAL Data access layer

iOS iPhone Operational System

UFRGS Universidade Federal do Rio Grande do Sul

CPD Centro de Processamento de Dados

SUMÁRIO

1 INTRODUÇÃO	12
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 Mobilidade Urbana	14
2.1.1 Carona	15
2.2 A plataforma iOS	15
2.3 Ferramentas	16
2.3.1 Xcode	17
2.3.2 git	17
2.3.3 Trello	18
2.4 Arquiteturas de <i>software</i>	18
2.4.1 Cliente-servidor	18
2.4.2 MVC	19
2.5 Padrões de projeto de software	20
2.5.1 <i>Repository</i>	20
2.5.2 <i>Singleton</i>	21
2.6 Requisitos	21
2.6.1 Funcionais	21
2.6.2 Não-funcionais	22
2.7 Metodologia de Desenvolvimento	22
2.8 <i>Framework</i>	23
2.8.1 Frameworks nativos	23
2.8.2 <i>Frameworks</i> de terceiros	24
2.9 Considerações Finais	24
3 TRABALHOS RELACIONADOS	25
3.1 Descrição dos Trabalhos	25
3.1.1 BlaBlaCar	25
3.1.2 Waze Carpool	25
3.1.3 Zumpy	26
3.2 Análise Comparativa	27
4 PROJETO UFRGS CARONAS	29
4.1 Elementos do sistema	29
4.2 Backend	30
4.3 Modelagem	30
4.4 Requisitos	31
4.5 Opções apenas mulheres	33
5 IMPLEMENTAÇÃO	34
5.1 Ambiente de desenvolvimento	34
5.2 Frameworks iOS utilizados	35
5.3 Alterações da modelagem	36
5.4 Padrões de projeto de software	36
6 O APLICATIVO	39
6.1 Onboarding	39
6.2 Tela de login	39
6.3 Modo Caronista	39
6.3.1 Busca de caronas	41
6.4 Modo Motorista	44
6.4.1 Cadastro de oferecimento de caronas	47
6.5 Perfil	54

7 AVALIAÇÃO COM USUÁRIOS	58
7.1 Ambientação dos experimentos	58
7.2 Protocolo de testes.....	59
7.2.1 Formulário Pré-testes	59
7.2.2 Apresentação do UFRGS Caronas.....	59
7.2.3 Etapa inicial	59
7.2.4 Etapa apenas para mulheres	60
7.2.5 Etapa final	60
7.3 Resultados.....	60
7.3.1 Perfil dos usuários	61
7.3.2 Análise do Teste de Usabilidade da Etapa Inicial.....	63
7.3.3 Análise do Teste de Usabilidade com mulheres	67
7.3.4 Análise do Teste de Usabilidade da Etapa Final.....	69
7.3.5 Sugestões.....	69
8 CONCLUSÕES	72
REFERÊNCIAS.....	75
APÊNDICE A — FORMULÁRIO PRÉ-TESTE	78
APÊNDICE B — FORMULÁRIO PÓS PRIMEIRA ETAPA	81
APÊNDICE C — FORMULÁRIO PÓS ETAPA APENAS COM MULHERES	83
APÊNDICE D — FORMULÁRIO PÓS ETAPA FINAL	85

1 INTRODUÇÃO

A mobilidade urbana é um tema que vem sendo muito debatido nos dias atuais. Com o aumento da quantidade de carros nas ruas, a fluidez do trânsito é muito prejudicada, o que afeta diretamente a percepção de qualidade de vida das pessoas que se locomovem diariamente nos grandes centros urbanos. Em Porto Alegre, a situação não é diferente. Na cidade, houve um crescimento de 70% na quantidade de carros nos últimos 10 anos (CARÚS, 2019).

No primeiro semestre de 2019, apenas considerando o contexto de Graduação, a UFRGS tinha 28.459 alunos matriculados, dentre os quais 27.267 tinham aulas presenciais em órgãos da cidade de Porto Alegre, segundo o Painel de Dados¹ da universidade. É seguro assumir que, hora ou outra, estas pessoas têm a necessidade de se locomover até algum dos campi da UFRGS para atender às aulas ou realizarem atividades como bolsa ou estágio, e assim, acabam por enfrentar os ônibus lotados e vias congestionadas.

Com isto em mente, a UFRGS, através do CPD² (Centro de Processamento de Dados), idealizou uma ferramenta que busca estimular e facilitar o compartilhamento de caronas solidárias, trazendo mais conforto para os usuários e ajudando, mesmo que pouco, a reduzir o trânsito na cidade. Este projeto foi denominado UFRGS Caronas. O público alvo do Projeto UFRGS Caronas, que foi decidido por ser implementado como um aplicativo móvel para as plataformas iOS e Android, são as pessoas que fazem parte da comunidade da UFRGS. O objetivo deste trabalho é a criação de um protótipo funcional de aplicativo para a plataforma iOS que implemente os requisitos do projeto UFRGS Caronas.

Ao final do trabalho, a usabilidade do protótipo obtido foi avaliada através de testes com usuários, onde foram definidas tarefas envolvendo as principais funcionalidades disponíveis no aplicativo. As avaliações foram analisadas e mostraram resultados positivos, o que levou o autor a crer que o protótipo desenvolvido possui boa usabilidade, através de uma interface intuitiva.

A motivação para esta proposta em específico se deu por o autor acreditar na ideia de que uma ferramenta que facilite e estimule a troca de caronas voluntárias no ambiente acadêmico poderia trazer benefícios para os indivíduos que dela participam, especialmente os alunos. Além disso, o projeto UFRGS Caronas já havia sido idealizado no CPD

¹<<https://www1.ufrgs.br/paineldedados>> Acesso em novembro de 2019. Ferramenta da UFRGS para visualização de dados a respeito da universidade

²<<https://www.ufrgs.br/cpd>> Acesso em outubro de 2019

da UFRGS (local onde o autor participou de bolsa de desenvolvimento durante a reta final de sua trajetória acadêmica), mas nunca foi implementado por completo.

O texto está estruturado de maneira a guiar o leitor através da jornada percorrida pelo autor no desenvolvimento deste trabalho. O Capítulo 2 apresenta os conceitos teóricos que fundamentaram a criação do protótipo e as tomadas de decisão necessárias para a definição das funcionalidades a serem implementadas. Em seguida, no Capítulo 3, são apresentadas ferramentas com objetivos similares já existentes no mercado. Estas soluções são então comparadas com a versão do protótipo obtida com este trabalho.

Em seguida, o Capítulo 4 dá um panorama sobre o contexto em que o aplicativo iOS desenvolvido neste trabalho se encontra dentro do Projeto UFRGS Caronas, descrevendo os elementos existentes e os esforços necessários para a conclusão deste trabalho. Com este plano de fundo, o Capítulo 5 descreve os detalhes da implementação do UFRGS Caronas, e em seguida, no Capítulo 6, é apresentado o protótipo resultante.

Então, no Capítulo 7, são descritos os experimentos feitos com usuários a fim de avaliar a usabilidade do UFRGS Caronas. Os resultados dos experimentos são então analisados, e em seguida, no Capítulo 8, o autor apresenta as conclusões às quais chegou no término deste trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste Capítulo, é apresentada a teoria que guiou a concepção e o desenvolvimento deste trabalho. Primeiramente, é dado um panorama a respeito do tema que o inspirou: Mobilidade Urbana. A seguir, são explanados os conceitos teóricos da computação utilizados no desenvolvimento do aplicativo UFRGS Caronas desenvolvido neste trabalho. O Capítulo é finalizado com as considerações finais.

2.1 Mobilidade Urbana

Mobilidade urbana é a facilidade encontrada em se deslocar dentro de um centro urbano, das mais diversas maneiras: de carro, ônibus, metrô, bicicleta, a pé, entre outras. Em (SILVA, 2017), o conceito é definido informalmente como “a movimentação de pessoas e coisas para dentro e para fora e para todos os cantos da cidade”. Isto porque mobilidade urbana não engloba apenas o fluxo de pessoas em um centro urbano, mas qualquer tipo de locomoção, como o transporte de cargas, por exemplo.

No Brasil especificamente, nota-se um aumento muito grande do número de carros nas ruas nas últimas décadas. Um fator fundamental para tal aumento foi a redução tributária e a iniciativa de investimento na indústria automobilística. Segundo (CARVALHO, 2016):

Com uma nova política de atração dos investimentos da indústria automobilística, iniciada em meados da década de 1990, o Brasil vem passando por outra fase de aumento do transporte individual motorizado. A capacidade de produção de automóveis e motocicletas mais que triplicou no período.

Mais carros nas ruas - cuja infra-estrutura não foi escalada na mesma proporção que o aumento no tráfego (CINTRA, 2013) - acaba gerando um trânsito mais pesado, e portanto, um deslocamento menos eficiente. Tendo em vista que a locomoção na cidade faz parte do cotidiano de seus habitantes, que todos os dias saem às ruas para trabalhar, estudar ou praticar atividades de lazer, a eficiência desse deslocamento acaba afetando diretamente na percepção de qualidade de vida dessas pessoas. Além disso, ainda há o fator ecológico envolvido, onde a poluição (atmosférica e sonora) e o gasto de energia cada vez maior são percebidos.

Por estes e outros fatores, cada vez mais tem-se discutido o tema e buscado soluções para melhorar o trânsito nas ruas das grandes metrópoles do Brasil e do mundo. Uma forma de diminuir um pouco a quantidade de carros nas ruas é o fomento à prática

de caronas (solidárias ou pagas), que aumenta a proporção entre passageiros por carro nas ruas, reduzindo a subutilização dos carros no trânsito. Esta prática virou tendência nos últimos tempos, e acarretou a criação de *startups* no ramo (como o aplicativo BlaBlaCar) e foi abraçada por empresas já consolidadas no mercado (como a Google, empresa dona do aplicativo Waze Carpool ¹).

2.1.1 Carona

A carona pode ser definida como uma forma de transporte que consiste no uso compartilhado de um automóvel particular, onde o proprietário cede espaço para outras pessoas, podendo ou não receber alguma espécie de retorno financeiro com este ato. Sob a ótica de mobilidade urbana, pode-se encarar esta prática como uma medida que visa a redução de viagens subutilizadas, uma vez que ocorre a ocupação de lugares até então vazios na viagem a ser realizada.

Considerando uma carona paga, como, por exemplo, a praticada em aplicativos como BlaBlaCar, fica claro que ambas as partes são beneficiadas: a pessoa que recebeu a carona viaja com maior conforto e por um valor provavelmente menor do que o praticado em outros serviços no mercado, e, quem ofereceu a carona, recebe um retorno financeiro para ceder um lugar em seu carro.

Mas e no caso das caronas solidárias? A pessoa que oferece a carona possui algum benefício ao ceder um lugar em seu carro sem receber dinheiro em troca? Estudos mostram que sim, pois a carona solidária pode contribuir na melhoria da autoestima, humor e qualidade de vida dos envolvidos (ARASAKI et al., 2016). No contexto deste trabalho, espera-se que as caronas solidárias possam gerar uma maior integração da comunidade da UFRGS.

2.2 A plataforma iOS

iOS² é o nome dado ao sistema operacional desenvolvido pela Apple exclusivamente para seus aparelhos móveis (iPhone, iPod Touch e iPad). Foi lançado em 2007, juntamente com o lançamento do primeiro iPhone, sob o nome de iPhone OS. Após isso, novas versões do sistema operacional foram lançadas juntamente com o lançamento anual

¹<<https://apps.apple.com/br/app/waze-carpool/id1091029104>> Acesso em outubro de 2019

²<<https://www.apple.com/br/ios>> Acesso em outubro de 2019

de modelos de iPhone. A partir da versão 4, o sistema deixou de se chamar iPhone OS, sendo abreviado para iOS. Atualmente, a versão atual é a 13, lançada em 3 de junho de 2019, na conferência anual de desenvolvedores da Apple (Apple Worldwide Developers Conference, ou WWDC³).

Para desenvolver aplicações voltadas para o sistema iOS, é necessário o uso de um computador rodando o sistema operacional proprietário de computadores da Apple, o macOS⁴, pois sem ele não é possível realizar o *build* da aplicação para a plataforma. A Apple recomenda o uso do software XCode: uma IDE com todo o ferramental para o desenvolvimento iOS, onde podem ser usadas as seguintes linguagens de programação:

- **Objective-C⁵**: foi criada no início da década de 1980 tendo como base a linguagem C, com o diferencial de adicionar suporte à programação orientada a objetos. Foi popularizada a partir de 1988, quando a NeXT⁶ comprou a licença da linguagem da StepStone⁷. Após realizar a compra da NeXT em 1996, a Apple passou a utilizar a linguagem em seus produtos, como no sistema operacional Mac OS X (recentemente renomeado macOS) e mais tarde, no iOS.
- **Swift⁸**: lançada em 2014, é uma linguagem de programação multiparadigma que se apresenta como uma linguagem moderna e focada em três aspectos: segurança, performance e suporte à aplicação de *design patterns* (SWIFT.ORG, 2019). Mesmo sendo consideravelmente recente, Swift demonstra uma comunidade de bom tamanho, ficando em 14^o nas linguagens mais populares e a 6^a mais amada segundo pesquisa em (Stack Overflow, 2018).

2.3 Ferramentas

Esta seção é destinada à explicação em detalhes das ferramentas utilizadas no desenvolvimento deste trabalho.

³<<https://developer.apple.com/wwdc19>> Acesso em outubro de 2019

⁴<<https://www.apple.com/br/macOS>> Acesso em outubro de 2019

⁵<<https://developer.apple.com/library/archive/documentation/Cocoa/Conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html>> Acesso em outubro de 2019

⁶Empresa norte-americana de hardware e software fundada em 1985 por Steve Jobs

⁷Antes chamada Productivity Products International (PPI), foi uma empresa de software fundada em 1983 por Brad Cox e Tom Love

⁸<<https://developer.apple.com/swift>> Acesso em outubro de 2019

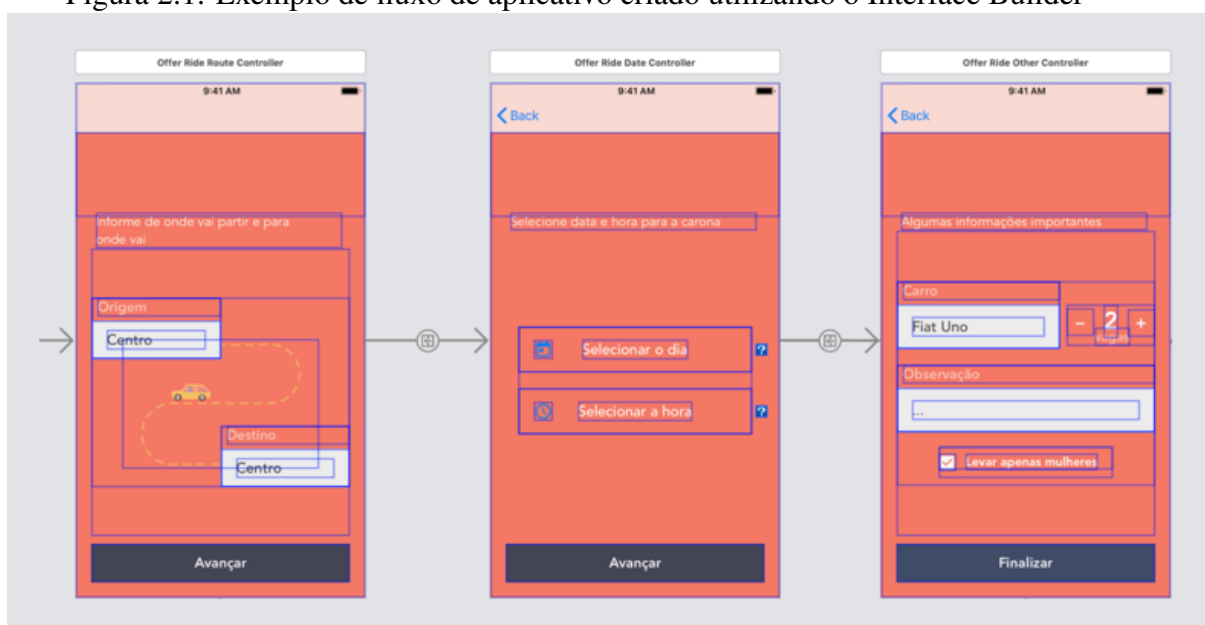
2.3.1 Xcode

O Xcode⁹ é a IDE fornecida pela Apple para uso em computadores com o sistema operacional macOS. O Xcode contém as ferramentas necessárias para o desenvolvimento de aplicativos para as plataformas proprietárias da Apple (macOS, iOS, iPadOS, watchOS e tvOS), desde a construção das telas e compilação do código-fonte até a geração do binário que mais tarde pode ser disponibilizado na App Store através do iTunes Connect (The Manifest, 2018).

O Xcode conta com o Interface Builder¹⁰, uma ferramenta de criação de interfaces que permite que o desenvolvedor crie as telas do aplicativo de forma gráfica e também defina as transições entre elas, facilitando na criação do fluxo do aplicativo.

A Figura 2.1 exemplifica parte do fluxo de um aplicativo criado utilizando o Interface Builder do XCode.

Figura 2.1: Exemplo de fluxo de aplicativo criado utilizando o Interface Builder



2.3.2 git

Ter um sistema de versionamento de código é fundamental em uma equipe de desenvolvimento de *software*. Tanto pela possibilidade de recorrer à uma versão anterior

⁹<<https://developer.apple.com/xcode>> Acesso em outubro de 2019

¹⁰<<https://developer.apple.com/xcode/interface-builder>> Acesso em outubro de 2019

do sistema caso algum problema seja observado, quanto pela documentação da evolução do projeto, que acaba sendo gerada ao longo do desenvolvimento.

O git¹¹ é o sistema de versionamento de código mais utilizado no mundo, passando em 2014 o Apache Subversion¹², segundo pesquisa descrita em (SKERRETT, 2014). Seu uso é gratuito e possui código aberto.

2.3.3 Trello

O Trello¹³ é um sistema de gestão visual estilo Kanban¹⁴ que permite a criação de quadros, listas e cartões para uma melhor organização de tarefas. É bastante utilizado no contexto de desenvolvimento de *software* (Polcode Team, 2018), devido à fácil visualização que sua forma de representação dos conceitos proporciona.

2.4 Arquiteturas de *software*

A arquitetura de um sistema define a estrutura básica que os módulos que o compõem devem seguir, e como devem se dar as relação entre eles. Um sistema pode, inclusive, conter mais de uma arquitetura, visto que cada camada da aplicação pode conter suas próprias regras de funcionamento. Nesta seção, são descritas as diferentes arquiteturas que foram utilizadas no UFRGS Caronas.

2.4.1 Cliente-servidor

A arquitetura Cliente-Servidor (BERSON, 1992) consiste de um sistema que possui aplicações servidoras que realizam tarefas requisitadas por aplicações clientes. Essencialmente, um cliente faz um pedido a um servidor e espera até receber uma resposta.

No contexto deste trabalho, a arquitetura Cliente-servidor se apresenta na comunicação entre o UFRGS Caronas, que funciona como cliente, e a API de dados, que é o servidor.

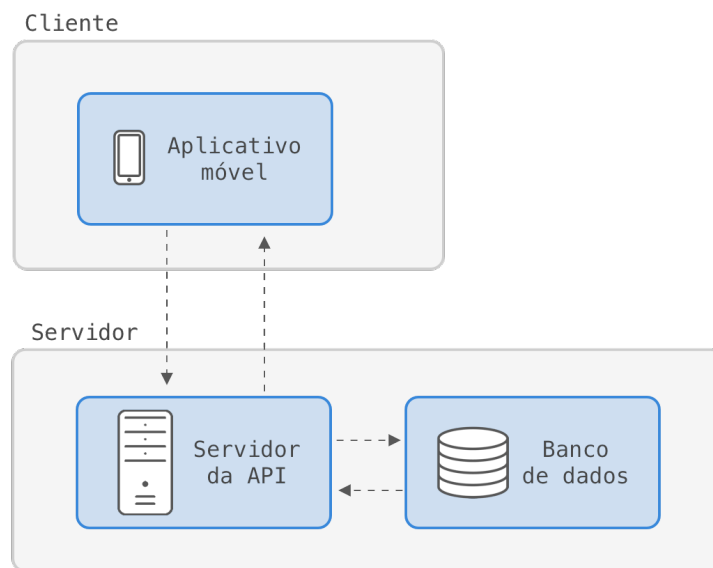
¹¹ <<https://git-scm.com>> Acesso em outubro de 2019

¹² <<https://subversion.apache.org>> Acesso em outubro de 2019

¹³ <<https://trello.com>> Acesso em outubro de 2019

¹⁴ <<https://artia.com/kanban>> Acesso em outubro de 2019

Figura 2.2: Arquitetura Cliente-servidor presente no sistema do UFRGS Caronas



2.4.2 MVC

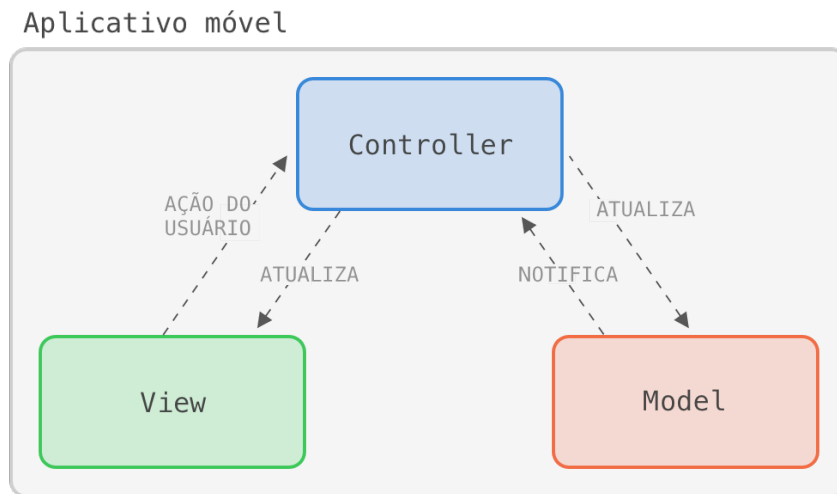
No aplicativo UFRGS Caronas, foi utilizado o padrão MVC (DEACON, 1995). Este é o padrão de arquitetura recomendado pela Apple para o desenvolvimento iOS, pois estimula a criação de código reutilizável e com interfaces de programação mais bem definidas (Apple Developer, 2018).

O padrão MVC é formado por três objetos principais, que são conceitualmente separados um do outro e cumprem cada um uma função específica (PERES; LASO-MARSETTI, 2019). São eles:

- **Model:** responsável por encapsular os dados da aplicação, bem como definir a lógica de negócio e as formas de operar sobre estes dados.
- **View:** é a camada responsável pela comunicação com o usuário. É através dela que são apresentados os dados, e capturados os comandos do usuário.
- **Controller:** é o mediador entre a Model e a View, atualizando esta com as novas informações daquela, e vice-versa.

A Figura 2.3 ilustra a comunicação entre os objetos do MVC descritos acima.

Figura 2.3: Arquitetura MVC presente no APP



2.5 Padrões de projeto de software

Nesta seção, são apresentados alguns dos padrões de projeto de *software* utilizados no desenvolvimento do UFRGS Caronas.

2.5.1 *Repository*

Repository (FOSTAINI, 2018), também chamado de repositório, é um padrão de projeto de *software* que permite o encapsulamento da lógica de acesso aos dados, separando-a da lógica de negócio da aplicação (camada de domínio).

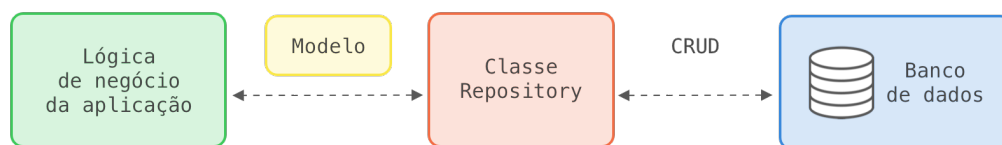
O uso deste padrão permite que a lógica de acesso aos dados em um banco de dados, por exemplo, seja modificada alterando apenas a implementação do repositório responsável, sem afetar a lógica de negócio da aplicação. Isto proporciona uma padronização na forma de acesso da aos dados, que se dá através do repositório. Seu funcionamento é como segue:

1. A aplicação faz uma requisição de dados para o repositório;
2. O repositório se comunica com o banco de dados através de operações CRUD (*Create, Read, Update e Delete*);
3. Quando receber do banco de dados os dados requisitados, o repositório os processa visando montar um objeto do modelo (entidade) requisitado;

4. O repositório retorna os dados obtidos para a aplicação.

A Figura 2.4 demonstra o funcionamento do padrão *Repository*, através de um exemplo onde uma aplicação se comunica com um banco de dados.

Figura 2.4: Fluxo da comunicação entre uma aplicação e um banco de dados usando o padrão *Repository*



2.5.2 Singleton

O padrão *Singleton* é utilizado quando se deseja que uma aplicação tenha apenas uma instância de uma certa classe, em todos os cenários possíveis (Lokesh Gupta, 2019). É necessário garantir que esta classe esteja disponível para todo o escopo da aplicação de forma única e que não seja possível instanciá-la novamente.

2.6 Requisitos

O conjunto de requisitos de um sistema geralmente é definido nas fases iniciais de um projeto de *software* e visa guiar a fase da implementação através da especificação do que o sistema deve fazer, os serviços que deve oferecer e as restrições ao funcionamento (SOMMERVILLE, 2011). Os requisitos são constantemente classificados em duas categorias: requisitos funcionais e requisitos não funcionais.

2.6.1 Funcionais

Um requisito funcional diz respeito a regras que o sistema deve seguir: serviços que deve fornecer, formas como deve reagir às diferentes entradas do usuário e o comportamento esperado em diferentes situações. Geralmente são específicos e dizem respeito apenas a uma parte do sistema.

2.6.2 Não-funcionais

Os requisitos não-funcionais podem dizer respeito às propriedades mais subjetivas do sistema, como confiabilidade, tempo de resposta e experiência do usuário, não estando diretamente relacionados com os serviços específicos que o sistema deve oferecer a seus usuários. Assim sendo, muitas vezes, os requisitos não-funcionais se aplicam ao sistema como um todo.

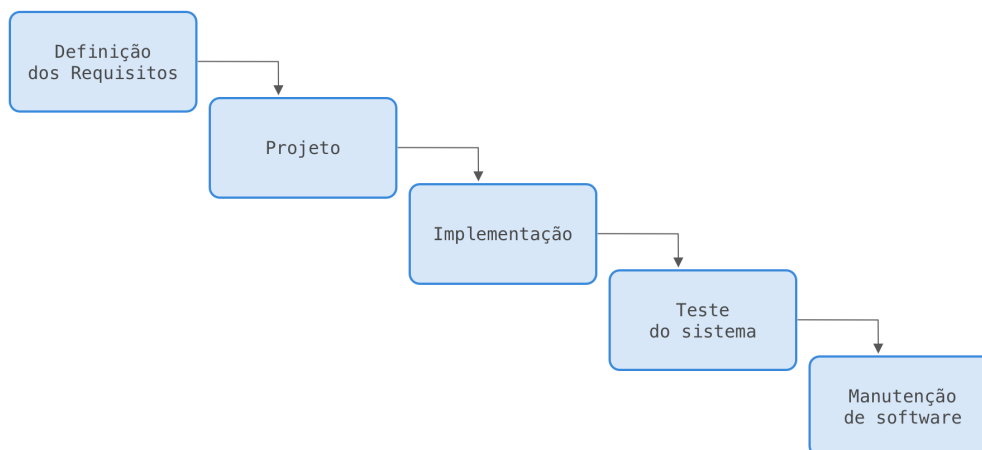
2.7 Metodologia de Desenvolvimento

A metodologia utilizada ao longo do desenvolvimento do UFRGS Caronas se assemelha bastante ao Modelo em Cascata. Este modelo foi escolhido pela necessidade de se definir muito bem os requisitos do sistema, e pela grande flexibilidade do projeto, que não necessitava de uma nova versão pronta do aplicativo a cada iteração.

O modelo em cascata (também chamado de *Waterfall*) faz uma abordagem sistemática e sequencial ao desenvolvimento do software (PRESSMAN, 1995), e é bastante enfático quanto ao planejamento e desenvolvimento das especificações (Altexsoft, 2016). Ele define 5 etapas pelas quais o projeto passa na ordem determinada, uma após a outra, como ilustra a Figura 2.5. As etapas são:

1. **Definição dos Requisitos:** é voltada para a definição dos objetivos do *software*, serviços que ele deve fornecer ao usuário e limitações que serão encontradas;
2. **Projeto:** é feito o planejamento que precede ao desenvolvimento de fato, incluindo definições como a arquitetura de *software* e as estruturas de dados que serão utilizadas;
3. **Implementação:** nesta etapa é onde acontece a codificação do sistema de fato, utilizando-se da documentação gerada nas etapas anteriores;
4. **Teste do sistema:** são investigados erros na implementação, buscando aumentar a segurança de que as entradas do sistema produzam resultados que coincidam com os requisitos especificados na primeira etapa;
5. **Manutenção de software:** por fim, esta etapa dedica-se a corrigir erros que não foram previamente detectados, bem como realizar melhorias funcionais no produto. É uma etapa do ciclo de vida do *software*, mas não pertence ao seu desenvolvimento propriamente dito.

Figura 2.5: Etapas do desenvolvimento de *software* segundo o Modelo Cascata



2.8 Framework

Um *framework* de *software* pode vir empacotado com a própria linguagem de programação, ou ser desenvolvido e fornecido por terceiros. A seguir estão listados os *frameworks* que foram utilizados no UFRGS Caronas.

2.8.1 Frameworks nativos

Esta seção é dedicada aos *frameworks* nativos (isto é, empacotados com o *Software Development Kit* do iOS) que foram utilizados na implementação do UFRGS Caronas.

- **Foundation**¹⁵: é o *framework* mais básico para o desenvolvimento iOS. Ele fornece os tipos de dados básicos, serviços do sistema operacional e outras funcionalidades essenciais para o programador, como implementação de *arrays*, armazenamento de dados e funções de rede;
- **UIKit**¹⁶: é o *framework* padrão para a criação de interface gráfica no desenvolvimento iOS. Ele fornece a infra-estrutura necessária para que o aplicativo mostre informações na tela para o usuário e capture eventos disparados a partir da interação com o sistema.

¹⁵ <<https://developer.apple.com/documentation/foundation>> Acesso em outubro de 2019

¹⁶ <<https://developer.apple.com/documentation/uikit>> Acesso em outubro de 2019

2.8.2 Frameworks de terceiros

A seguir estão listados os *frameworks* utilizados no UFRGS Caronas que foram desenvolvidos por terceiros.

- **Alamofire**¹⁷: fornece todo o ferramental para aplicações iOS realizarem comunicação através dos protocolos HTTP e HTTPS de maneira simples.
- **SwiftyJson**¹⁸: este *framework* busca simplificar a manipulação de objetos JSON em Swift. Esta tarefa não é muito simples sem o uso de alguma biblioteca, devido à forte tipagem da linguagem, que torna o código muito verboso neste caso;
- **SwiftOverlays**¹⁹: o uso deste *framework* tem uma função voltada para a experiência do usuário. Ele fornece várias maneiras de dar um *feedback* visual para o usuário enquanto o aplicativo realiza alguma tarefa que pode vir a demorar (por exemplo, a obtenção de dados no banco através da API);
- **Hero**²⁰: assim como o SwiftOverlays, o Hero é focado na interface com o usuário. Este *framework* fornece animações e transições de tela customizadas para tornar a experiência do usuário mais fluída e interessante.

2.9 Considerações Finais

Neste capítulo foram apresentados os conceitos que formam a base teórica para o desenvolvimento deste trabalho. Em suma, o objetivo é criar um aplicativo móvel que incentive a cooperação na comunidade acadêmica, e que por consequência, promova melhorias na mobilidade urbana da cidade.

Para atingir este objetivo, foram usadas duas arquiteturas de *software*: Cliente-Servidor na comunicação do aplicativo iOS com a API (que então se comunica com um banco de dados) e MVC na implementação do aplicativo em si.

¹⁷<<https://github.com/Alamofire/Alamofire>> Acesso em outubro de 2019

¹⁸<<https://github.com/SwiftyJSON/SwiftyJSON>> Acesso em outubro de 2019

¹⁹<<https://github.com/peterprokop/SwiftOverlays>> Acesso em outubro de 2019

²⁰<<https://github.com/HeroTransitions/Hero>> Acesso em outubro de 2019

3 TRABALHOS RELACIONADOS

Não é de hoje que vem se falando em melhorar o trânsito. Algumas iniciativas já foram tomadas. Neste capítulo, é dado um panorama a respeito de soluções existentes com o mesmo propósito ou similar à solução apresentada neste trabalho. Cada solução é descrita em detalhes, e em seguida, é feita uma análise comparativa entre elas e o presente trabalho.

3.1 Descrição dos Trabalhos

Nesta seção, estão descritas as seguintes aplicações: BlaBlaCar, Waze Carpool, Wunder e Zumpy.

3.1.1 BlaBlaCar

BlaBlaCar¹ é uma plataforma de caronas compartilhadas que possui interfaces mobile (iOS e Android) e web. Nela, o usuário pode publicar ofertas de caronas, bem como buscar caronas oferecidas por outros usuários. Ao cadastrar uma carona, o usuário escolhe um valor que cada caronista deverá pagar. Este valor é mostrado aos usuários que buscam caronas, e a ideia é que ele seja menor do que o valor de mercado da viagem, isto é, o preço de uma passagem de ônibus, transporte particular ou outros serviços.

A plataforma, que chegou ao Brasil em 2015, atingiu em agosto de 2019 a marca de 5 milhões de usuários no país (BlaBlaCar, 2019). E a expectativa da empresa é de crescimento ainda maior nos próximos anos, devido a uma tendência crescente de consumidores que cada vez mais repensam a compra de um carro próprio.

3.1.2 Waze Carpool

O Waze² é um aplicativo móvel de navegação bastante consolidado no mercado, tendo cerca de 100 milhões de usuários no mundo todo (Waze for Brands, 2019). Em

¹<<https://apps.apple.com/br/app/blablacar/id341329033>> Acesso em outubro de 2019

²<<https://apps.apple.com/us/app/waze-navigation-live-traffic/id323229106>> Acesso em outubro de 2019

2013, a plataforma, mantida por uma startup israelense, foi comprada pela gigante Google (Google, 2013).

Em 2016, a empresa lançou o aplicativo Waze Carpool, com a proposta de conectar motoristas com rotas parecidas no trajeto de casa para o trabalho. Inicialmente, apenas a região da Baía de São Francisco (Califórnia, USA) foi contemplada com o serviço (HAWKINS, 2016), que então foi se espalhando pelos Estados Unidos e o mundo. O aplicativo chegou ao Brasil em agosto de 2018, chamando a atenção do público para o baixo valor das caronas, que no primeiro mês custariam R\$ 2 para todos os usuários (O Globo, 2018).

3.1.3 Zumpy

O Zumpy³ é um *app* brasileiro criado em 2015 com o mesmo intuito dos demais: promover a troca de carona entre pessoas com rotas parecidas. A diferença, porém, é que no Zumpy, os motoristas passam por um cadastro prévio na plataforma, necessitando inclusive de um segundo aplicativo para a função de motorista (Zumpy, 2019). Inicialmente, as caronas eram solidárias. Atualmente, o modelo de negócios mudou e as caronas passaram a ser pagas e só acontecem com um mínimo de 2 passageiros⁴.

A relação entre passageiro e motorista, bem como o fato de as caronas serem pagas, aproximam o Zumpy de outros aplicativos de transporte como Uber e Cabify, que são serviços de transporte particular com motoristas presentes na localização do usuário. O Zumpy, porém, não busca ser a fonte de renda principal dos motoristas, mas sim uma renda extra nas rotas que ele já faria no seu cotidiano.

Um ponto positivo do aplicativo é que o fato de as caronas recebidas serem agendadas previamente e terem uma rota pré-estabelecida simplifica o processo para o usuário, uma vez que não se faz necessário buscar as viagens compartilhadas todos os dias. Além disso, teoricamente é gerada uma sensação de confiança para com a imagem do motorista, visto que este é o mesmo em todas as caronas.

³<<https://apps.apple.com/us/app/zumpy-transporte-compartilhado/id979190956>> Acesso em outubro de 2019

⁴<<https://www.zumpy.com.br>> Acesso em outubro de 2019

3.2 Análise Comparativa

Nesta seção, é feita uma análise de diversos aspectos relevantes das aplicações direcionadas à mobilidade urbana, comparando as soluções descritas na seção anterior com o presente trabalho. Esta comparação é feita com o auxílio de 3 tabelas, que possuem dados coerentes com a situação atual de cada solução na data presente de escrita deste trabalho (outubro de 2019).

A tabela 3.1 lista as plataformas em que cada uma das soluções se encontra disponível. É possível notar que nenhuma das soluções possui versão *Desktop* e todas possuem versões iOS e Android, com exceção do UFRGS Caronas, que no presente momento possui apenas a versão iOS implementada.

Tabela 3.1: Plataformas em que a aplicação está disponível

	BlaBlaCar	Waze Carpool	Zumpy	UFRGS Caronas
<i>Web</i>	Sim	Sim	Não	Não
<i>Mobile iOS</i>	Sim	Sim	Sim	Sim
<i>Mobile Android</i>	Sim	Sim	Sim	Não
<i>Desktop</i>	Não	Não	Não	Não

A tabela 3.2 trata das funcionalidades disponíveis para o usuário em cada uma das soluções. As funcionalidades mais básicas do domínio das aplicações, como “Oferecer carona” e “Pedir carona” estão presentes em todas as soluções. O UFRGS Caronas, neste caso, se destaca por não ter implementadas: algumas funcionalidades que podem ser consideradas mais avançadas, como “Avaliação do motorista” e “Chat”; outras que não se encaixam no escopo do aplicativo, como “Cobrança por carona”; e também funcionalidades básicas de aplicativos em geral, como “Notificações *push*”, que é vista pelo autor como trabalho futuro.

Tabela 3.2: Funcionalidades disponíveis

	BlaBlaCar	Waze Carpool	Zumpy	UFRGS Caronas
Oferecimento de caronas	Sim	Sim	Sim	Sim
Busca de caronas com filtros	Sim	Sim	Não	Sim
Pedido de carona	Sim	Sim	Sim	Sim
Aceite/recusa de pedidos de carona	Sim	Sim	Sim	Sim
Cobrança por carona	Sim	Sim	Sim	Não
Avaliação do motorista	Sim	Sim	Sim	Não
Notificações <i>push</i>	Sim	Sim	Sim	Não
<i>Chat</i>	Sim	Sim	Sim	Não

Na tabela 3.3, são tratadas questões relacionadas à segurança do usuário na utilização de cada uma das soluções. É possível ver que o UFRGS Caronas segue a tendência crescente de preocupar-se em especial com a segurança de usuárias mulheres, que são mais afetadas pela violência no cotidiano devido aos hábitos da sociedade machista na qual vivemos. No UFRGS Caronas, assim como nas outras plataformas, é disponibilizado às usuárias mulheres a escolha de oferecer caronas apenas para outras mulheres, bem como pegar caronas de outras mulheres apenas.

Ainda na questão de segurança, o UFRGS Caronas é o único que conta com o *login* vinculado ao cadastro do usuário na universidade. Ao utilizar este cadastro, há a garantia de apenas haver caronas oferecidas por pessoas da comunidade da UFRGS, o que teoricamente aumenta a sensação de segurança dos usuários.

Tabela 3.3: Opções de segurança

	BlaBlaCar	Waze Carpool	Zumpy	UFRGS Caronas
Opção para caronas apenas com mulheres (disponível apenas para mulheres)	Sim	Sim	Sim	Sim
<i>Login</i> vinculado ao cadastro na universidade	Não	Não	Não	Sim
Cadastro especial para motoristas	Não	Sim	Sim	Não

4 PROJETO UFRGS CARONAS

O UFRGS Caronas é um projeto projetado no CPD da UFRGS, que tem o objetivo de disponibilizar para a comunidade da UFRGS um aplicativo (para as plataformas iOS e Android) que faça o intermédio entre pessoas que vão de carro para a universidade e estão dispostas a oferecer caronas solidárias (podendo-se combinar informalmente uma contribuição por parte do caronista) e pessoas que façam um caminho parecido e se interessem em pegar caronas para ir ou voltar de algum campus da UFRGS.

Antes do início do desenvolvimento do presente trabalho, o projeto já possuía a infraestrutura descrita em (CONCEICAO, 2017), no formato de uma API de dados. O aplicativo móvel, porém, nunca chegou a ser implementado em nenhuma das plataformas móveis.

Este capítulo busca dar um panorama geral do funcionamento do sistema, deixando claro o que havia sido feito previamente ao início deste trabalho, e descrever o levantamento de requisitos feito para o desenvolvimento do UFRGS Caronas.

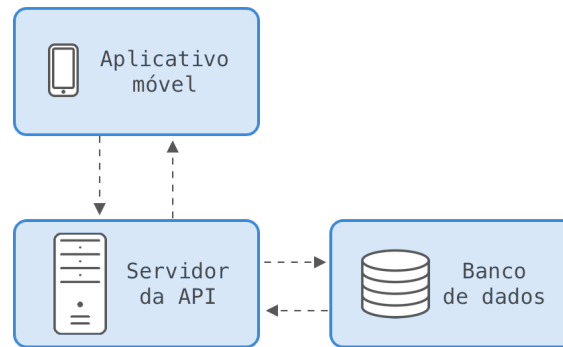
4.1 Elementos do sistema

O ecossistema completo do projeto UFRGS Caronas é - atualmente - composto dos seguintes elementos:

- **Aplicativo iOS:** aplicativo móvel para a plataforma iOS, desenvolvido neste trabalho;
- **API:** *web service* cuja base foi desenvolvida em (CONCEICAO, 2017) e sofreu pequenas modificações para se adequar ao aplicativo.
- **Banco de dados:** onde são persistidas as informações do sistema. A implementação do banco de dados é visível apenas à API, e transparente ao aplicativo.

No projeto UFRGS Caronas, o aplicativo iOS desenvolvido faz requisições HTTPS para a API de dados, que responde também através do protocolo HTTPS com dados no formato JSON. A API é o módulo responsável pela comunicação com o banco de dados. A Figura 4.1 ilustra o funcionamento geral do projeto UFRGS Caronas.

Figura 4.1: Comunicação entre os elementos do projeto UFRGS Caronas



4.2 Backend

Em um sistema computacional, *Backend* (NEWMAN, 2015) é o nome dado ao módulo que lida com tarefas que não são visíveis ao usuário final, mas são fundamentais para o bom funcionamento do sistema como um todo. Enquanto, por exemplo, um aplicativo móvel é considerado o *frontend* (lado mais próximo do usuário) de um sistema, os servidores que lidam com bancos de dados, requisições e computações mais pesadas compõem o *backend*.

No projeto UFRGS Caronas, o *backend* foi implementado anteriormente ao início do desenvolvimento deste trabalho, e está documentado em (CONCEICAO, 2017), e ele é a parte do projeto responsável por fornecer os dados para o usuário final que utilizará o aplicativo iOS desenvolvido neste trabalho.

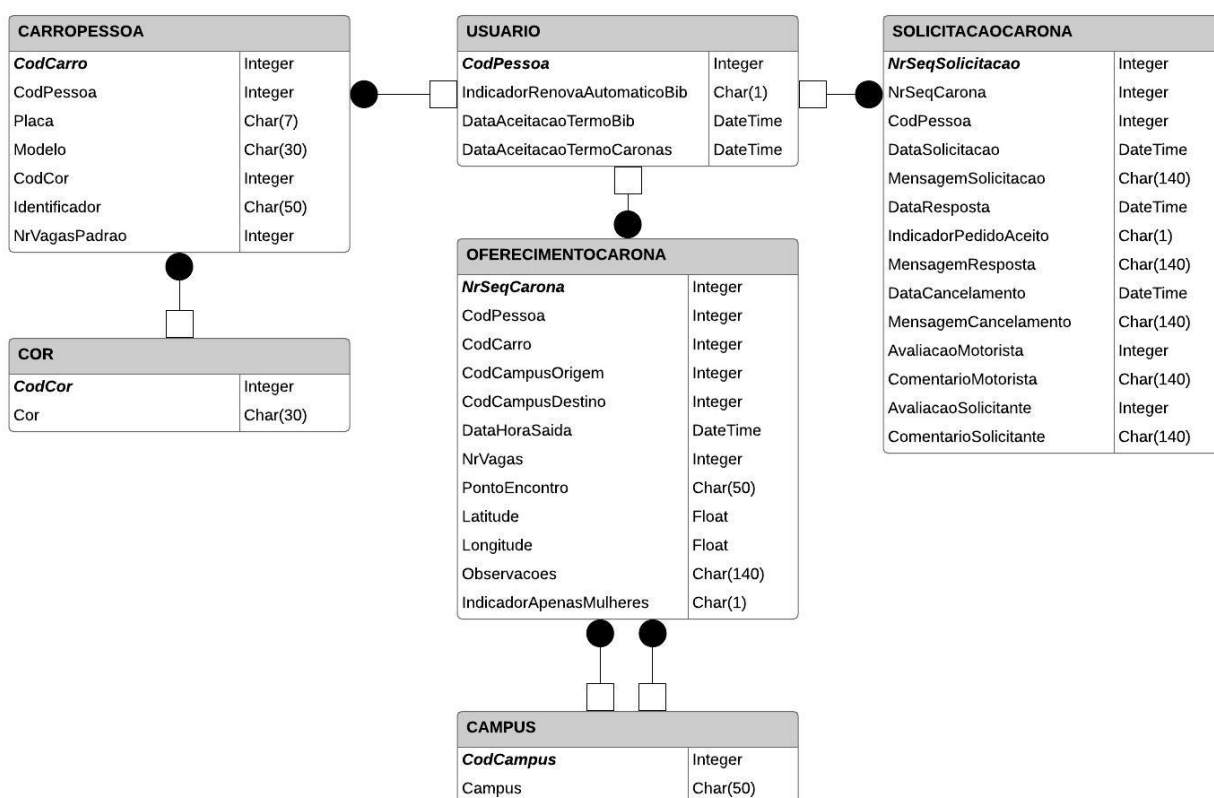
4.3 Modelagem

Nesta seção, é apresentada a modelagem original do projeto UFRGS Caronas, isto é, como foi definida durante a implementação do *web service* explicado em detalhes em (CONCEICAO, 2017). A Figura 4.2 mostra os modelos utilizados, que são os seguintes:

- **Campus** - associa cada campus da Universidade a um código numérico;
- **CarroPessoa** - contém dados de um carro e identificação de seu proprietário;
- **OferecimentoCarona** - contém dados sobre uma oferta de carona, incluindo a identificação do criador do oferecimento, o carro utilizado e o ponto de encontro dos usuários;
- **Cor** - associa cores a códigos numéricos;

- **SolicitacaoCarona** - contém dados sobre uma solicitação de carona, incluindo o criador da solicitação, o oferecimento à qual esta solicitação se refere e uma possível mensagem ao criador do oferecimento;
- **UsuarioCaronas** - contém dados referentes ao uso de *front-ends* ligados ao módulo Caronas, como uma variável que indica o aceite ou não dos termos de uso. Dados do usuário em si, como nome e identificação numérica da UFRGS, são implementados no modelo Pessoa herdado do núcleo da API.

Figura 4.2: Modelagem original do UFRGS Caronas



4.4 Requisitos

Esta seção é dedicada à listagem dos requisitos levantados para a implementação do UFRGS Caronas. Os requisitos funcionais levantados para o UFRGS Caronas são:

1. O UFRGS Caronas deve permitir *login* ao usuário com as credenciais da universidade (cartão do usuário na UFRGS e senha do portal da universidade);
2. O usuário deve estar logado para ter acesso a toda e qualquer funcionalidade do

UFRGS Caronas;

3. O UFRGS Caronas deve permitir que o usuário cadastre um ou mais carros;
4. O UFRGS Caronas deve permitir que o usuário edite os carros cadastrados previamente;
5. O UFRGS Caronas deve permitir que o usuário exclua os carros cadastrados previamente;
6. O UFRGS Caronas deve permitir que o usuário cadastre oferecimentos de caronas;
7. O UFRGS Caronas deve permitir que o usuário exclua oferecimentos de caronas cadastrados previamente;
8. O UFRGS Caronas deve permitir que o usuário realize consultas sobre o banco de caronas cadastradas, podendo aplicar zero ou mais filtros de busca;
9. O UFRGS Caronas deve permitir que o usuário crie um pedido de carona para os oferecimentos resultantes da busca, podendo ou não enviar uma mensagem;
10. O UFRGS Caronas deve permitir que o usuário aceite um pedido feito para seu oferecimento de carona, podendo ou não mandar uma mensagem de resposta;
11. O UFRGS Caronas deve permitir que o usuário recuse um pedido feito para seu oferecimento de carona, podendo ou não mandar uma mensagem de resposta;
12. O UFRGS Caronas deve permitir que o usuário veja o *status* dos pedidos de carona que realizou;
13. O UFRGS Caronas deve permitir que o usuário veja os oferecimentos de carona que cadastrou.
14. O UFRGS Caronas deve permitir que usuárias mulheres ofereçam caronas apenas para outras mulheres.
15. O UFRGS Caronas deve permitir que usuárias mulheres busquem por caronas oferecidas por outras mulheres.

Já os requisitos não-funcionais levantados para o UFRGS Caronas são:

1. O UFRGS Caronas deve ter uma interface intuitiva;
2. O UFRGS Caronas deve funcionar bem nas diversas versões de iPhone às quais estará disponível;
3. O UFRGS Caronas deve ser eficaz na comunicação com a API através de qualquer que seja a rede de internet disponível;
4. O UFRGS Caronas deve permitir três tipos de usuário: alunos, professores e servi-

dores da UFRGS.

4.5 Opções apenas mulheres

O projeto UFRGS Caronas conta com funcionalidades que estão disponíveis apenas para usuárias mulheres, seguindo uma tendência que vem sendo observada em diversos serviços no ramo de transporte particular, como os trabalhos citados no capítulo 2, o aplicativo 99¹ e outros (GNIPPER, 2019). Esta iniciativa visa oferecer maior segurança para mulheres, as maiores vítimas de violência doméstica e sexual no Brasil (GOMES, 2019), através da opção de restringir o contato da usuária do serviço apenas com outras mulheres. No UFRGS Caronas, são duas as funcionalidades disponíveis apenas para mulheres:

- Oferecimento de carona apenas para outras mulheres: neste caso, pessoas cujo gênero registrado no banco de dados da UFRGS não seja “Feminino” não terão acesso à carona oferecida.
- Busca de carona de usuárias mulheres: ao buscar uma carona, a usuária pode adicionar um filtro que faz com que sejam mostradas apenas caronas oferecidas por mulheres.

¹<<https://99app.com>> Acesso em outubro de 2019

5 IMPLEMENTAÇÃO

Este capítulo trata dos detalhes da implementação do UFRGS Caronas, apresentando as ferramentas utilizadas, alterações necessárias na infra-estrutura pré-existente ao início do projeto e como foram empregadas técnicas de programação que auxiliaram no desenvolvimento do trabalho.

5.1 Ambiente de desenvolvimento

Esta seção descreve os detalhes do ambiente em que foi realizado o trabalho, abordando desde o *hardware* utilizado até as ferramentas que auxiliaram na organização das tarefas.

Foram utilizadas dois computadores para o desenvolvimento do UFRGS Caronas:

- Macbook Pro mid-2014 com sistema operacional macOS versão 10.14.5
- iMac late-2013 com sistema operacional macOS versão 10.13.6

Para a escrita e compilação do código, bem como a criação das telas do UFRGS Caronas, foi utilizado o Xcode na versão 10.1. Sua ferramenta de construção de *layout* e fluxo das telas do aplicativo, o Interface Builder, foi bastante utilizado no desenvolvimento deste trabalho.

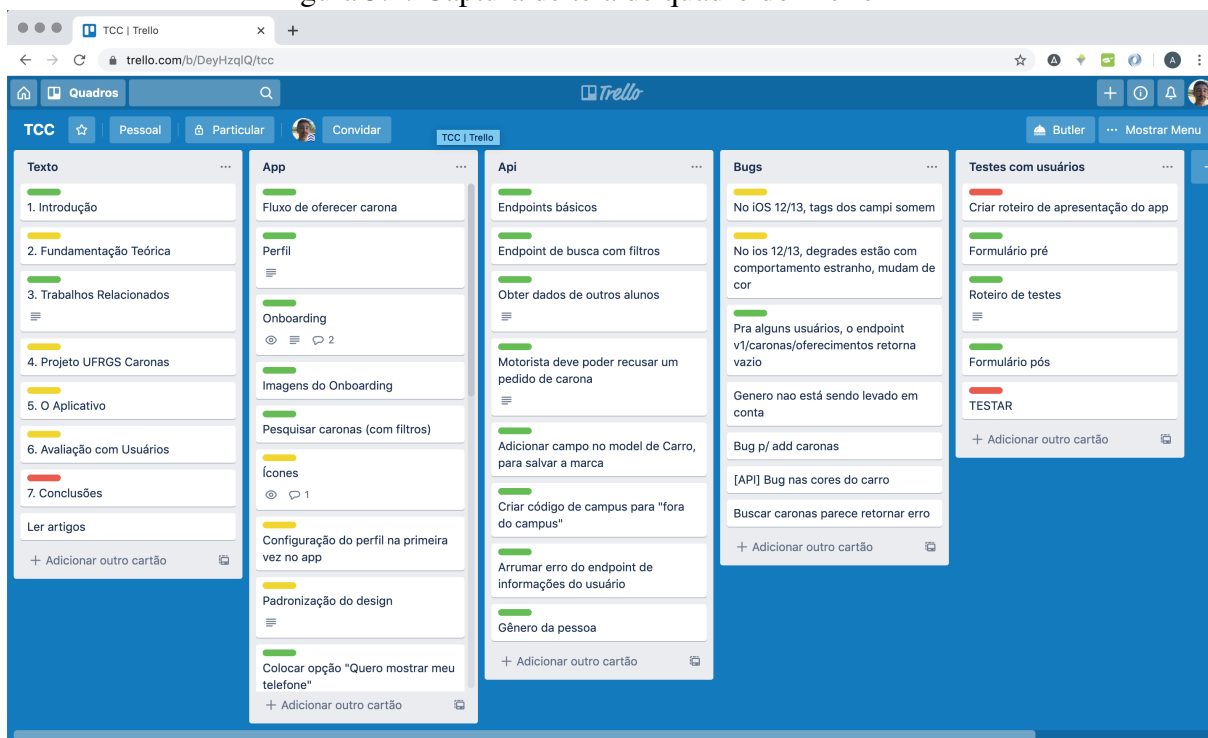
No que diz respeito à linguagem de programação, foi escolhida Swift na versão 5.1. A escolha do Swift, em detrimento de ObjC, se deu por maior experiência do autor com a linguagem, e por motivos citados anteriormente, como o fato de ser uma linguagem muito mais expressiva do que sua alternativa.

Para melhor organização e versionamento do código, foi utilizada a ferramenta git, através da criação de um repositório no Github ¹.

Por fim, para auxiliar a organização das tarefas durante o desenvolvimento do UFRGS Caronas, foi utilizado o Trello, através da criação de um quadro com listas destinadas a cada uma das supertarefas que compuseram este trabalho. A Figura 5.1 mostra o quadro descrito.

¹<<https://github.com/gutoboranga/ufrgs-caronas>> Acesso em outubro de 2019

Figura 5.1: Captura de tela de quadro do Trello



5.2 Frameworks iOS utilizados

Foram utilizados os seguintes *frameworks* no UFRGS Caronas:

- **Foundation:** foi utilizado pois é o padrão e fornece funcionalidades básicas como tipos de dados primitivos e gerenciamento de memória;
- **UIKit:** foi utilizado para a criação da interface gráfica do aplicativo. É um *framework* padrão no iOS;
- **Alamofire:** foi utilizado para realizar todas as chamadas feitas à API de dados;
- **SwiftyJson:** foi utilizado para manipular dados retornados da API, em formato JSON, de maneira mais simples;
- **SwiftOverlays:** foi utilizado para indicar ao usuário o carregamento de algum dado, como uma chamada à API, por exemplo.
- **Hero:** foi utilizado nas transições das telas e animações mais complexas.

5.3 Alterações da modelagem

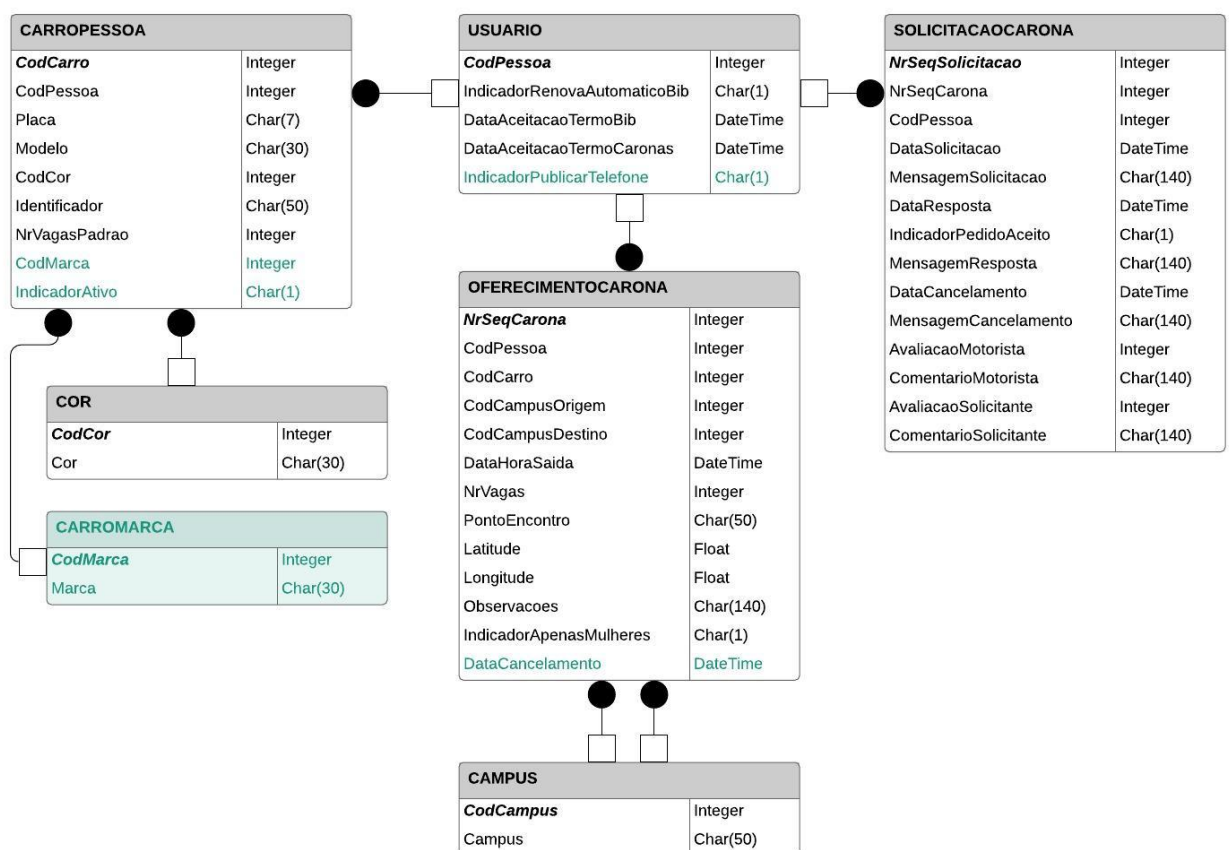
Ao longo do desenvolvimento deste trabalho, ocorreu a necessidade da criação de novas tabelas no banco de dados e da adição de campos em tabelas já existentes na modelagem original do *web service* do UFRGS Caronas. A seguir estão listadas as alterações, e na Figura 5.2, as adições à modelagem estão demarcadas na cor verde.

- **Tabela CARROMARCA** - foi criada uma nova tabela no banco de dados, responsável por associar cada marca de carro disponível para o usuário no cadastro de um novo carro a um código numérico. Possui dois campos:
 - *CodMarca* - código numérico da marca. É a chave primária;
 - *Marca* - nome da marca, do tipo *String*.
- **Campo *CodMarca* na tabela CARROPESSOA** - Foi criado um campo numérico que indica, no modelo de um carro, a marca à qual ele está associado.
- **Campo *IndicadorAtivo* na tabela CARROPESSOA** - foi criado um campo do tipo *char* que indica se um carro está ativo ou não. Quando um usuário exclui um carro, este é marcado como inativo, e não é mais relacionado na lista de carros do usuário, acontecendo a exclusão lógica do mesmo.
- **Campo *IndicadorPublicarTelefone* na tabela USUARIO** - foi criado um campo do tipo *char* que indica se o usuário deseja que seu telefone cadastrado seja mostrado a usuários que forem lhe dar carona, ou a quem ele for dar carona.
- **Campo *DataCancelamento* na tabela OFERECIMENTOCARONA** - foi criado um campo do tipo *DateTime* que, se estiver preenchido, indica a data em que ocorreu o cancelamento do oferecimento de carona associado. O cancelamento pode ocorrer quando o usuário criador do oferecimento de carona o exclui ou quando o carro associado com o oferecimento é excluído.

5.4 Padrões de projeto de software

Neste trabalho, o padrão repositório é utilizado em diversas ocasiões em que é necessária a obtenção e/ou criação de instâncias de uma entidade no banco de dados. Por exemplo, para criar no banco de dados uma instância da classe *Car* (que representa um carro de um usuário), a aplicação chama o método *create* da classe *CarRepository*,

Figura 5.2: Modelagem atualizada do UFRGS Caronas



que implementa a comunicação com a API e fará as requisições necessárias para a criação do registro no banco de dados.

O padrão *Singleton* é utilizado na classe `CarBrandRepository`, que é um repositório contendo a lista de marcas dentre as quais o usuário pode escolher para cadastrar seu carro na plataforma. Esta lista é obtida fazendo-se uma requisição para a API, e então mantida em memória. É importante que em qualquer momento que esta lista for acessada, ela tenha sempre os mesmos valores. Por este motivo, a implementação do padrão *Singleton* foi escolhida, garantindo que apenas uma instância da classe será utilizada em toda a aplicação.

6 O APLICATIVO

Este capítulo é dedicado a apresentar o resultado deste trabalho, descrevendo as funcionalidades presentes no UFRGS Caronas.

Ao abrir o UFRGS Caronas pela primeira vez, o usuário se depara com o *onboarding*, que consiste em duas telas explicando o propósito do aplicativo. Quando avança, é mostrada a tela de *login*, onde o usuário insere suas credenciais para usar as funcionalidades do aplicativo. Após logar-se no UFRGS Caronas, o usuário entra no fluxo principal do aplicativo, que consiste em três abas: *Modo Caronista*; *Modo Motorista*; *Perfil*. Inicialmente, a tela selecionada é a de *Modo Caronista*.

As seções a seguir explicam mais a fundo sobre cada uma das telas com as quais o usuário interage no UFRGS Caronas.

6.1 Onboarding

A primeira coisa que o usuário vê ao abrir o UFRGS Caronas pela primeira vez é o *onboarding*, que consiste em uma sequência com duas telas de boas-vindas explicando brevemente o objetivo do aplicativo. A Figura 6.1 mostra as telas que compõem o *onboarding* do UFRGS Caronas.

6.2 Tela de login

Após passar pelo *onboarding*, é apresentada ao usuário a tela de *login*. Ela possui dois campos de texto onde o usuário deve inserir, respectivamente, seu número de usuário no sistema da universidade, e sua senha. A Figura 6.2 mostra a tela de *login*.

6.3 Modo Caronista

Esta tela expõe ao usuário as informações e funcionalidades relativas ao papel de caronista no UFRGS Caronas. Nela, são mostrados os últimos pedidos de carona realizados pelo usuário, bem como o *status* deste pedido, que pode ser: Aceito, Recusado, Pendente ou Cancelado. Além disso, é mostrado um botão que leva o usuário à tela de busca de caronas, como mostra a Figura 6.3.

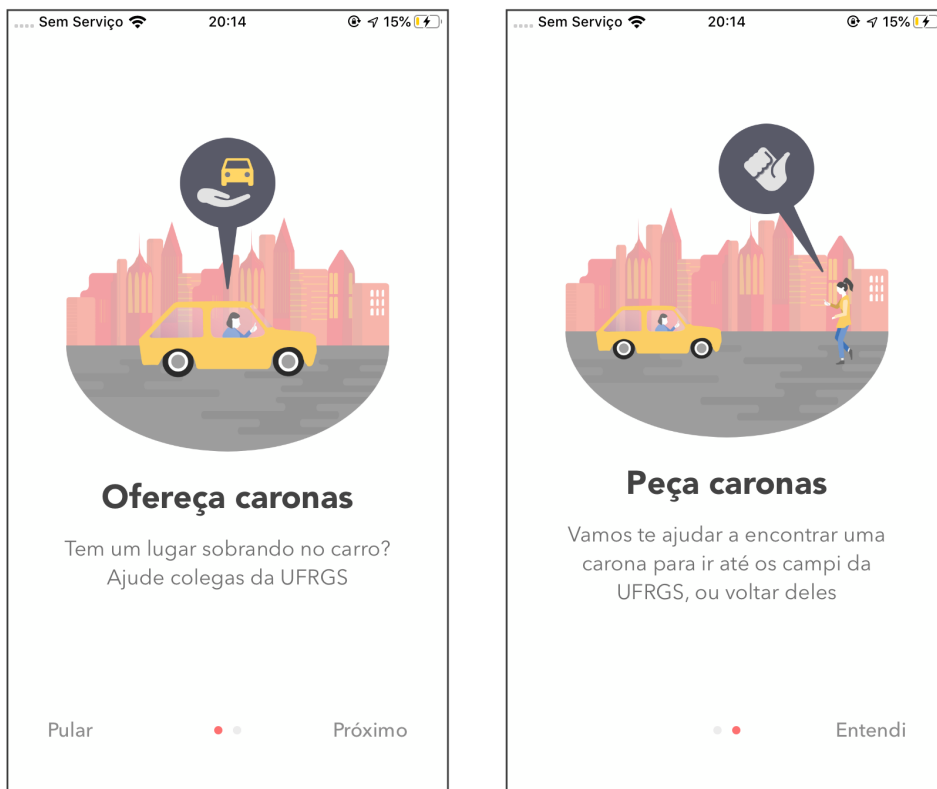
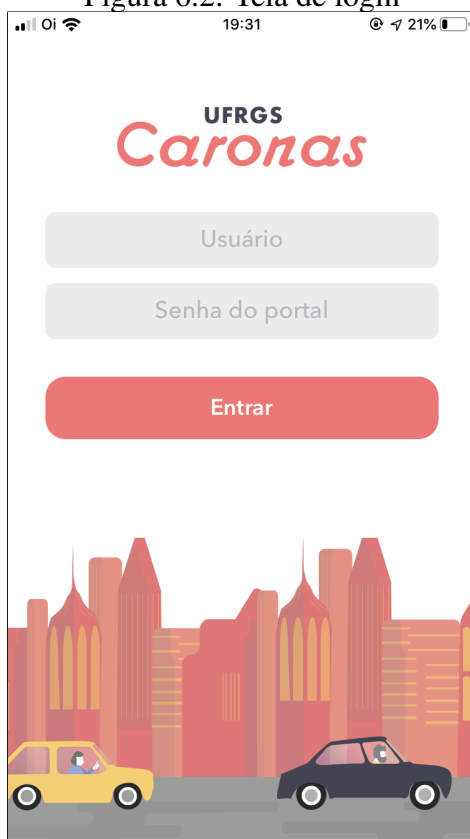
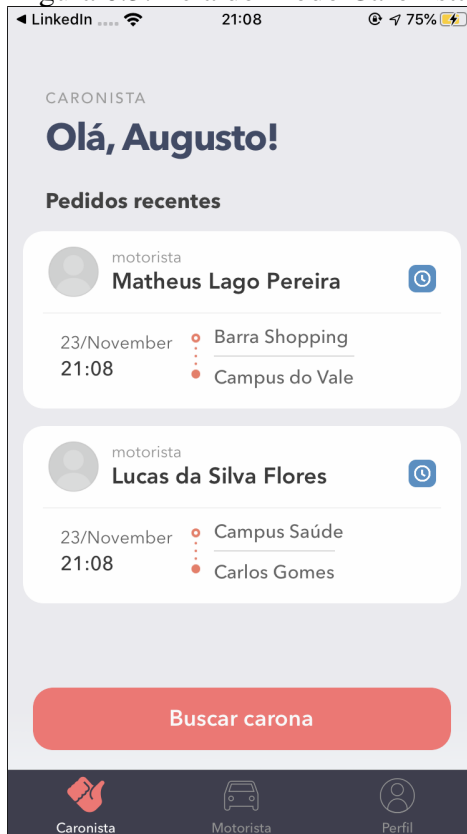
Figura 6.1: *Onboarding* do UFRGS Caronas

Figura 6.2: Tela de login



Na tela de Modo Caronista, o usuário pode selecionar um item da lista de pedidos de caronas. Então, é mostrada uma tela com os detalhes da carona, bem como o *status* do pedido de carona feito pelo usuário, como a Figura 6.4 ilustra.

Figura 6.3: Tela do Modo Caronista



6.3.1 Busca de caronas

No fluxo de Busca de Caronas, o usuário pode buscar por caronas oferecidas por outros usuários. É possível adicionar filtros com relação à origem e ao destino da carona (como mostra a Figura 6.5). Se o usuário for uma mulher, é habilitada a opção “Apenas mulheres”, onde serão mostradas apenas caronas que foram cadastradas por usuárias mulheres (e atendam aos outros filtros de localidade selecionados), como mostra a Figura 6.6.

Após adicionar filtros (ou limpá-los), o UFRGS Caronas faz a busca na API e apresenta os resultados para o usuário. A Figura 6.7 mostra um exemplo de tela onde a busca não gerou resultados. Na Figura 6.8, por sua vez, há resultados para a busca realizada.

Figura 6.4: Tela com os detalhes da carona e o *status* do pedido, que neste caso é “Pendente”



Figura 6.5: Tela de filtros para a busca de caronas



Figura 6.6: Filtro booleano disponível apenas para usuárias mulheres

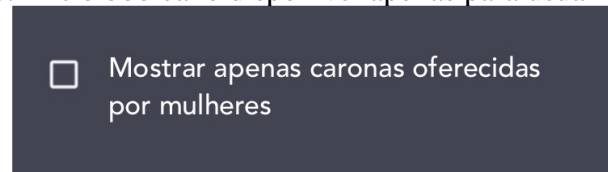


Figura 6.7: Tela de busca de caronas quando não há resultados para a busca



Figura 6.8: Tela mostrada quando a busca gerou resultados



Ao selecionar uma carona do resultado de sua busca, o usuário é levado à tela que mostra os detalhes do oferecimento de carona selecionado, como mostra a Figura 6.9. Além disso, nesta tela é possível fazer a criação de um pedido de carona, podendo ser adicionada uma mensagem para o motorista (como mostra a Figura 6.10).

Se o pedido for feito com sucesso, é mostrada a tela representada na Figura 6.11, que indica ao usuário que o pedido foi feito. Caso contrário, o usuário é alertado do erro ocorrido.

6.4 Modo Motorista

Esta é a tela que reúne as funcionalidades relativas ao papel de motorista no UFRGS Caronas. É nela que o usuário vê a lista dos oferecimentos de caronas que criou (como mostra a Figura 6.12), e também pode cadastrar novos oferecimentos.

Ao selecionar um item da lista de oferecimentos, o usuário é levado a uma tela que mostra os detalhes do oferecimento de carona selecionado, como mostra a Figura 6.13.

Na tela dos detalhes do oferecimento de carona, o usuário também pode visualizar

Figura 6.9: Tela com detalhes da carona resultante da busca



Figura 6.10: Tela onde o usuário adiciona uma mensagem para o motorista

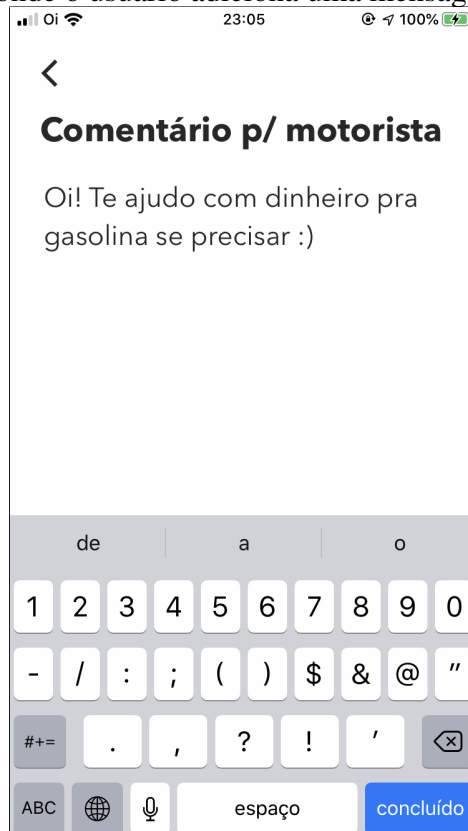


Figura 6.11: Tela que mostra pro usuário que o pedido da carona foi feito com sucesso



Figura 6.12: Tela que mostra a lista de oferecimentos de carona criados pelo usuário

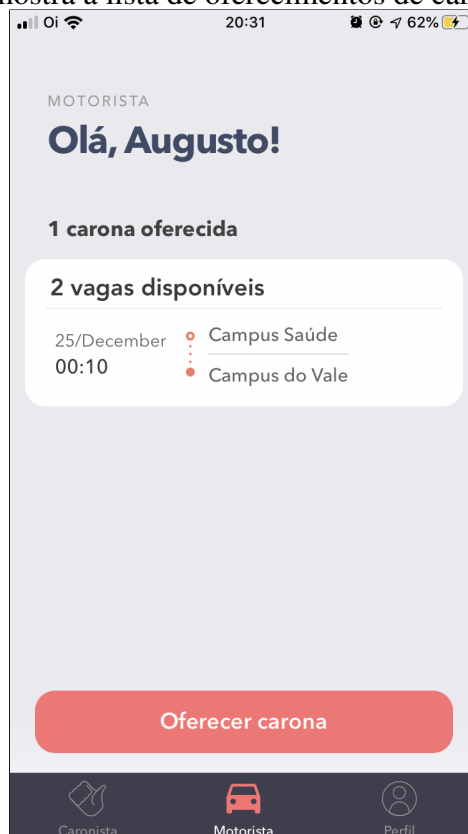


Figura 6.13: Tela que mostra os detalhes da carona oferecida



os pedidos que a carona recebeu. A Figura 6.14 mostra um caso onde o oferecimento de carona possui três pedidos respondidos: dois aceitos e um recusado. Na Figura 6.15, é mostrado um caso onde há um pedido ainda não respondido. Neste caso, o usuário pode tocar no pedido para respondê-lo, como mostra a Figura 6.16.

6.4.1 Cadastro de oferecimento de caronas

Ao tocar no botão “Oferecer carona”, o usuário é levado para o fluxo de oferecimento de carona, onde ele passa por três telas nas quais preencherá as informações relativas à carona oferecida. As etapas são:

1. **Escolha dos locais de origem e destino:** o primeiro passo é a escolha da origem e do destino da carona, como mostra a Figura 6.17. A interface possui dois botões, um para a escolha da origem e outro para o destino. Ao clicar em um dos dois botões, é apresentada uma tela, representada na Figura 6.19, com as possíveis localizações para que o usuário escolha. No caso de seleção da origem, é pedido ao usuário que descreva o ponto de encontro da carona (conforme a Figura 6.18).

Figura 6.14: Tela que mostra um oferecimento de carona com solicitações respondidas (aceitas e recusadas)



Figura 6.15: Tela com novas solicitações de carona



Figura 6.16: Tela onde o usuário responde a um pedido de caronas

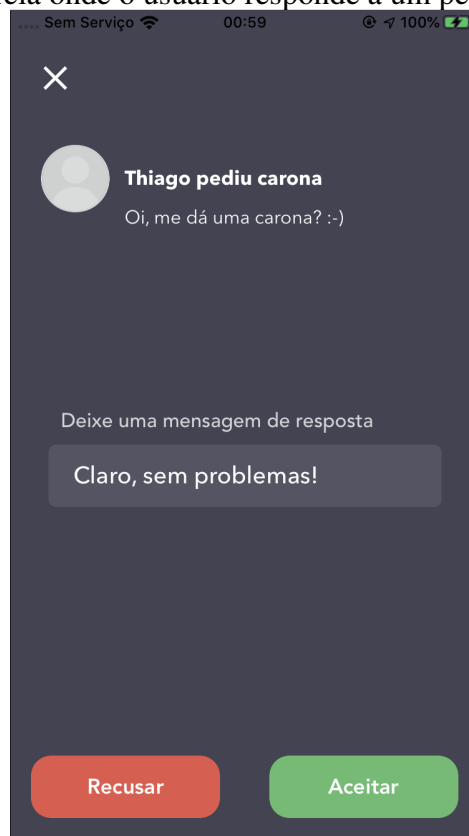


Figura 6.17: Telas antes e depois da escolha da origem e do destino da carona

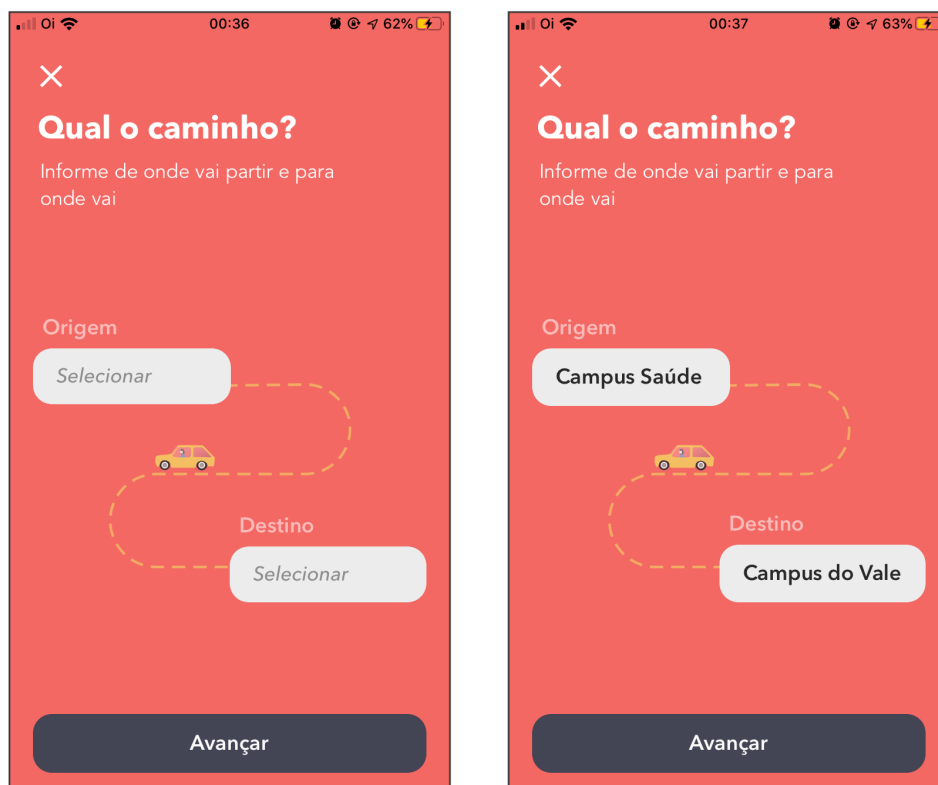


Figura 6.18: Tela onde o usuário informa detalhes do ponto de encontro da carona

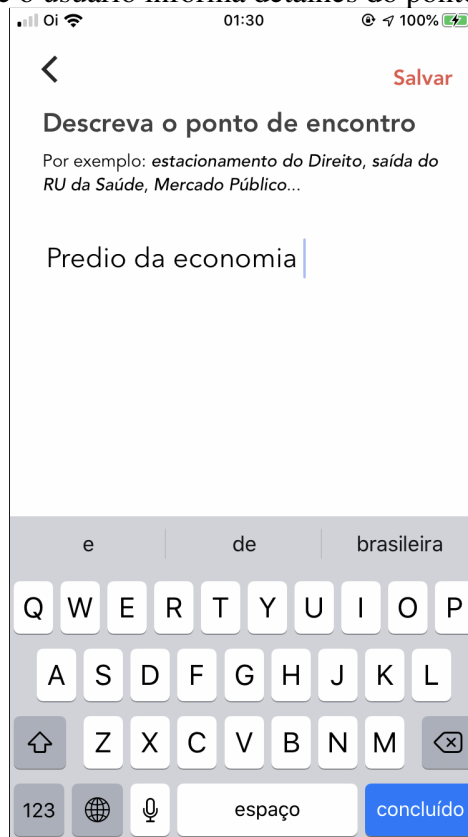
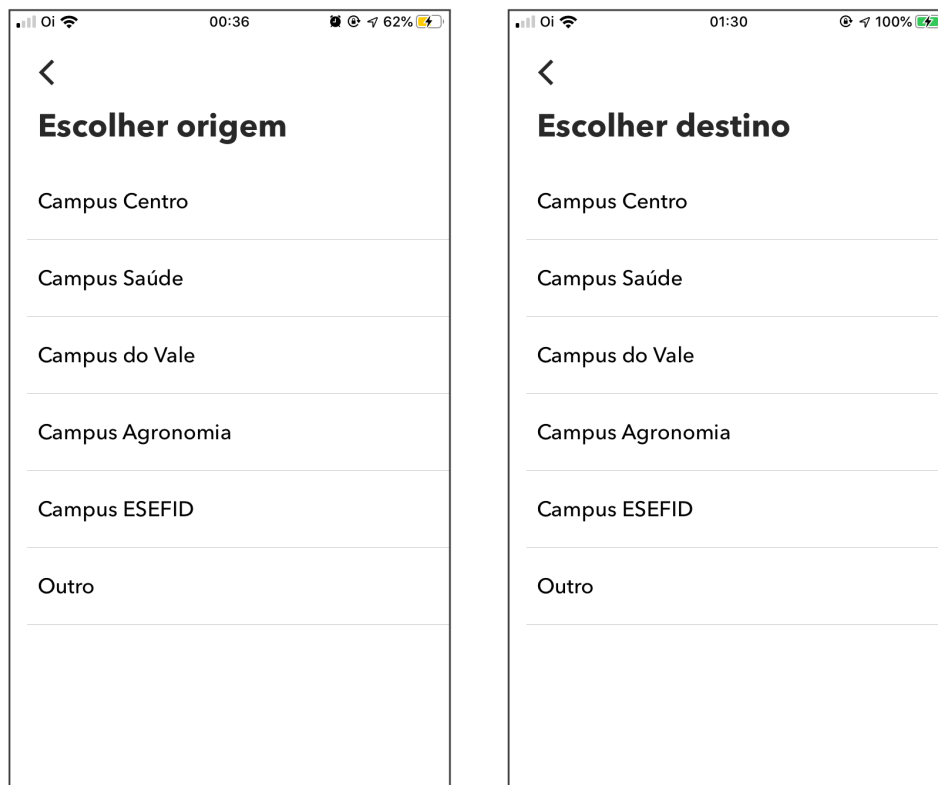
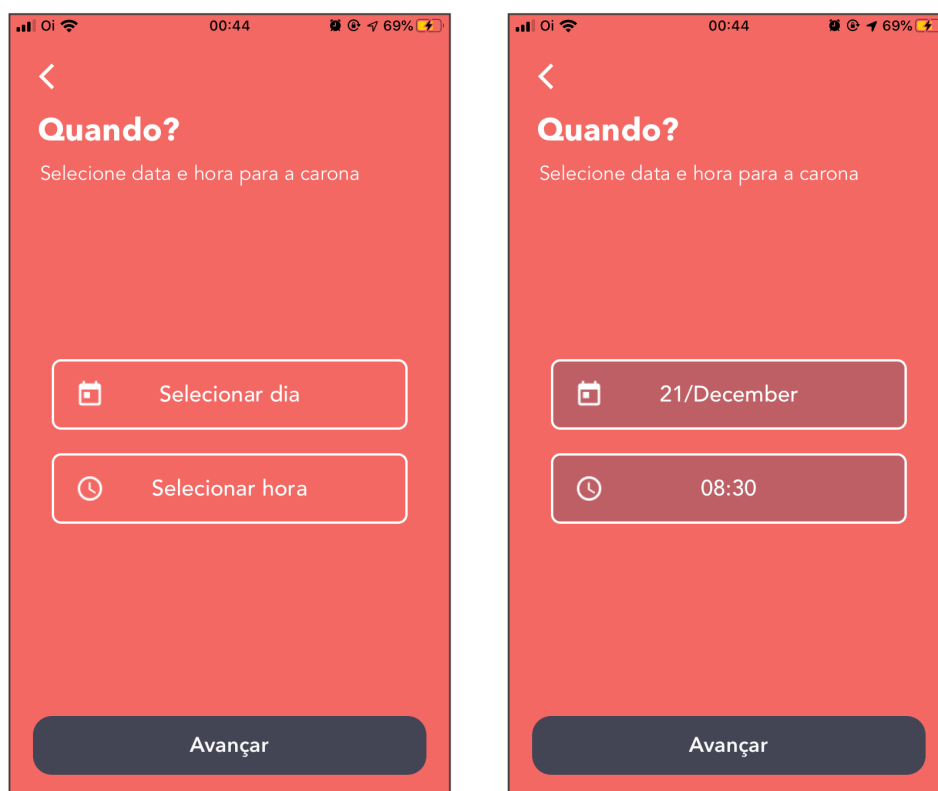


Figura 6.19: Telas onde o usuário escolhe o local de origem e de destino da carona, respectivamente



- Escolha da data e horário da carona:** em seguida, acontece a escolha da data e horário em que a carona ocorrerá. A interface, mostrada na Figura 6.20, possui dois botões que, se tocados, fazem abrir uma janela de seleção (como mostra a Figura 6.21) para a escolha da data (dia e mês) e do horário (hora e minuto), respectivamente.

Figura 6.20: Telas antes e depois da escolha da data e horário da carona



- Escolha de outros detalhes:** por fim, o usuário define os últimos detalhes da carona, como representado na Figura 6.22. São eles: o carro utilizado para a carona; o número de vagas disponíveis e alguma observação extra, caso queira. A Figura 6.23 mostra as telas de seleção do carro e adição de observações do oferecimento de carona. Para usuárias mulheres, além dos campos citados acima, é mostrada uma opção (conforme a Figura 6.24 ilustra) que faz com que o oferecimento de carona, quando finalizado, seja mostrado apenas para outras mulheres.

Se o oferecimento de carona for feito com sucesso, é mostrada para o usuário uma tela com essa mensagem, como mostra a Figura 6.25.

Figura 6.21: Janelas de seleção da data e horário da carona



Figura 6.22: Última tela de cadastro de carona, com escolha do carro, observações e número de vagas

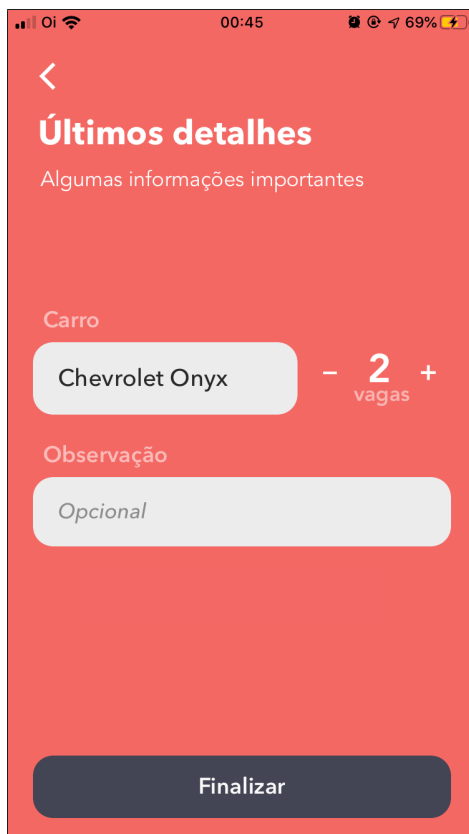


Figura 6.23: Telas de escolha de carro para a carona e adição de observações, respectivamente

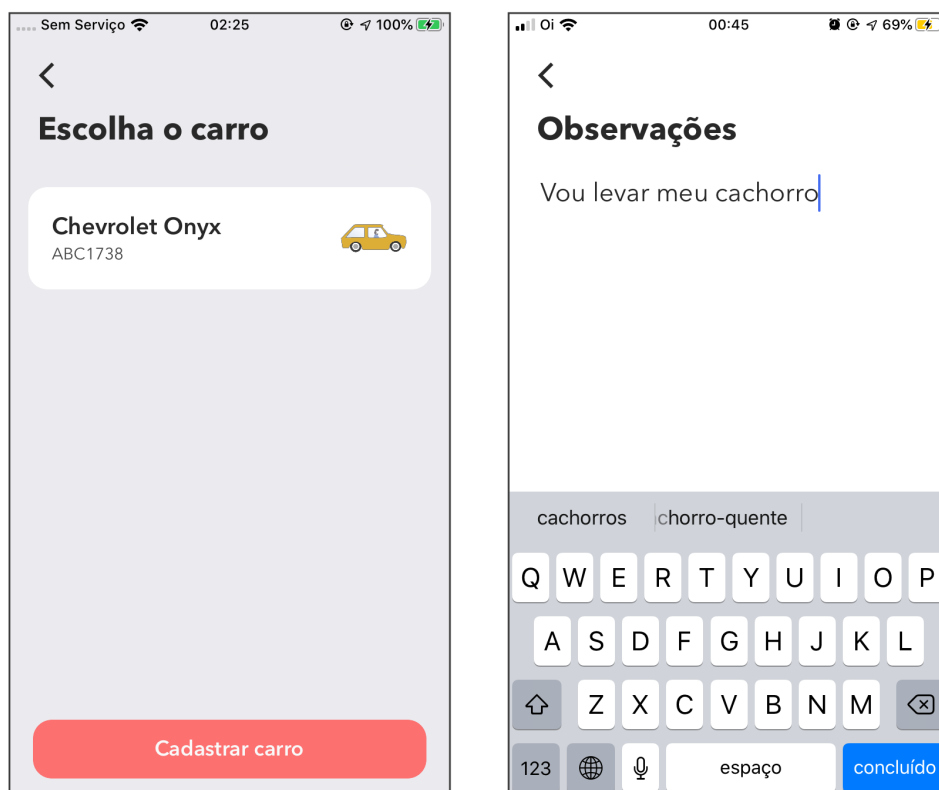


Figura 6.24: Opção disponível para usuárias mulheres, que quando selecionada indica que a carona sendo cadastrada deverá ser vista apenas por outras mulheres

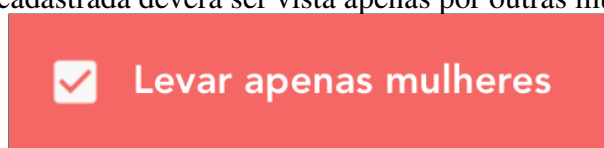


Figura 6.25: Tela que indica sucesso ao cadastrar a carona



6.5 Perfil

Na tela de perfil (mostrada na Figura 6.26) são apresentadas as informações do usuário, como nome e foto, além de opções para visualizar os carros cadastrados e o histórico de pedidos de carona realizados ou caronas oferecidas, conforme mostram as Figuras 6.27 e 6.28, respectivamente.

Ao selecionar a opção “Gerenciar carros”, o usuário é direcionado à tela que mostra a lista de carros cadastrados pelo usuário, como mostra a Figura 6.29. O usuário pode, então, selecionar um carro para editar ou excluir (conforme Figura 6.30), ou então cadastrar um novo carro (vide Figura 6.31).

Figura 6.26: Perfil do usuário



Figura 6.27: Histórico de caronas oferecidas pelo usuário



Figura 6.28: Histórico de caronas solicitadas pelo usuário



Figura 6.29: Tela que mostra a lista de carros do usuário

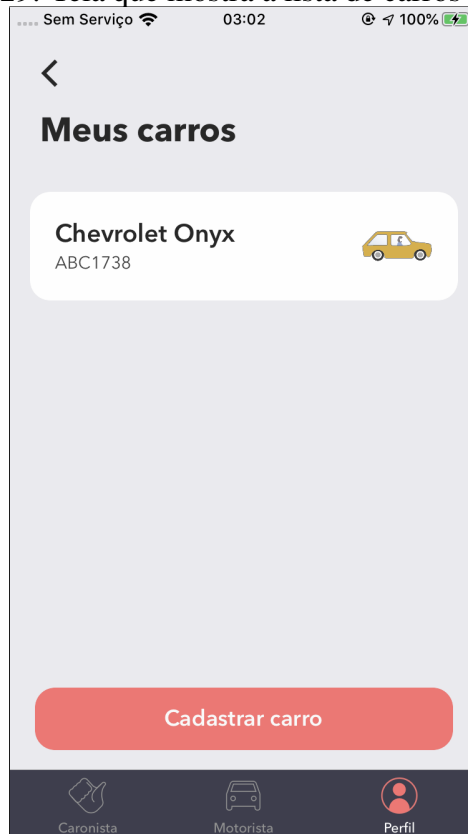


Figura 6.30: Tela de edição de um carro do usuário

Sem Serviço 03:03 100%

< Editar carro

Chevrolet


Modelo

Onyx

Placa

ABC1738

Cor

 Rosa

Atualizar

Caronista Motorista Perfil

Figura 6.31: Tela de cadastro de um novo carro

Oi 00:45 70%

× Cadastrar carro

Marca

Escolher marca


Modelo

Digite o modelo

Placa

Ex.: ABC0000

Cor

 Preto

Adicionar

7 AVALIAÇÃO COM USUÁRIOS

Este capítulo é dedicado à apresentação e análise dos experimentos realizados com usuários que aceitaram participar voluntariamente. A avaliação consistiu em um Teste de Usabilidade baseado em uma sequência de tarefas pré-determinadas que visavam testar a usabilidade das funcionalidades implementadas no UFRGS Caronas.

Para a avaliação de cada tarefa, foram apresentadas ao usuário 5 opções balanceadas, que vão de "Muito difícil" a "Muito fácil", seguindo a Escala Likert (LIKERT, 1932).

7.1 Ambientação dos experimentos

O experimento foi realizado com a presença do autor, pois o aplicativo ainda não estava disponível para *download*. Por este motivo, os experimentos ocorreram em diferentes locais, dependendo da combinação feita entre mim e os participantes. Todos estes locais contavam com acesso à internet e ambientes com ausência de grandes interferências que comprometessem a concentração da pessoa participante do teste. Os locais em que as avaliações ocorreram foram:

- Minha casa;
- CPD da UFRGS.

Em algumas instâncias do teste foi utilizada a rede Wi-fi disponível no local do teste, e em outras, a rede móvel (3G) do aparelho celular, visando testar um cenário de uso real, com o usuário empunhando o *smartphone* na rua. Foi utilizado um iPhone 7 com sistema operacional iOS versão 13.1.3 para a aplicação dos testes.

Usuários que possuem credenciais válidas na universidade as utilizaram para realizar *login* no UFRGS Caronas, conforme orientação do autor. Para pessoas que não possuem vínculo com a universidade, foram disponibilizadas credenciais "falsas", que, dentro do aplicativo, foram substituídas pelas credenciais do autor. Foram fornecidos dois tipos de credenciais falsas: uma para pessoas do gênero masculino e outra para pessoas do gênero feminino. Esta separação foi feita pelo fato de que nestes casos, o aplicativo não tinha a informação do gênero da pessoa testando (usada para mostrar, ou não, as *features* exclusivas para mulheres), uma vez que as credenciais reais eram as do autor.

7.2 Protocolo de testes

A seguir estão descritas todas as etapas do teste com usuários, e suas respectivas tarefas. Cada participante do experimento cumpriu todas as tarefas na ordem especificada, de acordo com sua aptidão para tal, conforme será explicado.

7.2.1 Formulário Pré-testes

O primeiro passo das pessoas testando o UFRGS Caronas é ler um termo de consentimento em que são apresentados os riscos e o usuário declara que sua participação na avaliação é voluntária. A avaliação só é continuada se o usuário selecionar a opção declarando que aceita participar. O apêndice A mostra este formulário pré-teste.

Após, caso a pessoa tenha aceitado participar do teste, é solicitado que ela responda um questionário informando seus dados básicos, assim como informações de sua experiência com o uso de internet, uso de *smartphone* e aplicativos de caronas.

7.2.2 Apresentação do UFRGS Caronas

A primeira parte da rotina de teste com cada pessoa voluntária é apresentar o UFRGS Caronas e suas funcionalidades, fazendo um passo a passo para situá-la acerca do contexto e do objetivo do aplicativo.

7.2.3 Etapa inicial

Em seguida, ocorre a primeira etapa em que a pessoa realizará tarefas utilizando o aplicativo. Esta etapa é obrigatória para todos, e nela são executadas algumas tarefas introdutórias, como a realização de *login* no UFRGS Caronas, e outras mais avançadas, como o cadastro de caronas. A lista completa de tarefas é a seguinte:

1. Logar-se no aplicativo com as credenciais da universidade;
2. Cadastrar um carro;
3. Cadastrar um oferecimento de carona;
4. Buscar por caronas saindo do campus Centro;

5. Fazer uma solicitação para uma carona resultante da busca.

Após a execução, a pessoa avalia sua experiência na realização das tarefas executadas nesta etapa, através de perguntas no formulário apresentado no apêndice B.

7.2.4 Etapa apenas para mulheres

Esta etapa é executada apenas com pessoas que se declararem do gênero feminino no Formulário Pré-testes, pois seu objetivo é testar funcionalidades do UFRGS Caronas que estarão disponíveis apenas para usuárias mulheres. As tarefas a serem executadas são:

1. Cadastrar um oferecimento de carona apenas para mulheres;
2. Buscar por caronas apenas para mulheres.

Após a execução, a pessoa avalia sua experiência na realização das tarefas executadas nesta etapa, através de perguntas no formulário do apêndice C.

7.2.5 Etapa final

Esta é a etapa final dos testes, onde são executadas tarefas de *logout* do sistema ou que dependam de tarefas realizadas previamente na sequência do teste. As tarefas são:

1. Excluir um oferecimento de carona.
2. Deslogar-se do aplicativo

Após a execução, a pessoa avalia sua experiência na realização das tarefas executadas nesta etapa, respondendo perguntas no formulário que está apresentado no apêndice D.

7.3 Resultados

Nesta seção são apresentados os resultados dos testes aplicados com usuários. Primeiramente, é traçado perfil do usuário, usando os dados obtidos no formulário pré-testes. Em seguida, são analisados os resultados das avaliações feitas pelos usuários sobre cada funcionalidade testada no aplicativo. Por fim, serão apresentadas as sugestões e

observações que os usuários deram a respeito do UFRGS Caronas.

7.3.1 Perfil dos usuários

Foram realizados testes com 13 pessoas no total, dentre as quais, 12 possuem algum tipo de vínculo com a UFRGS (5 estudantes e 7 servidores). Como é possível ver na Figura 7.1, do total de pessoas participantes, 11 se declararam do gênero masculino, e 2 do gênero feminino. No que se refere à idade dos participantes, as respostas foram bastante variadas, dentro do intervalo de 23 a 41 anos. O gráfico na Figura 7.2 representa a distribuição da idade dos participantes.

Figura 7.1: Gráfico dos gêneros declarados pelos participantes da avaliação com usuários

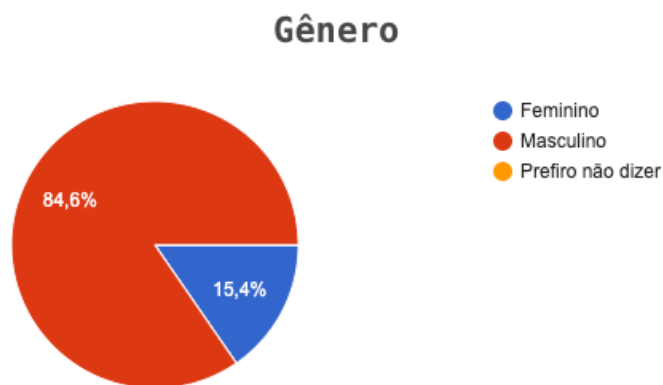
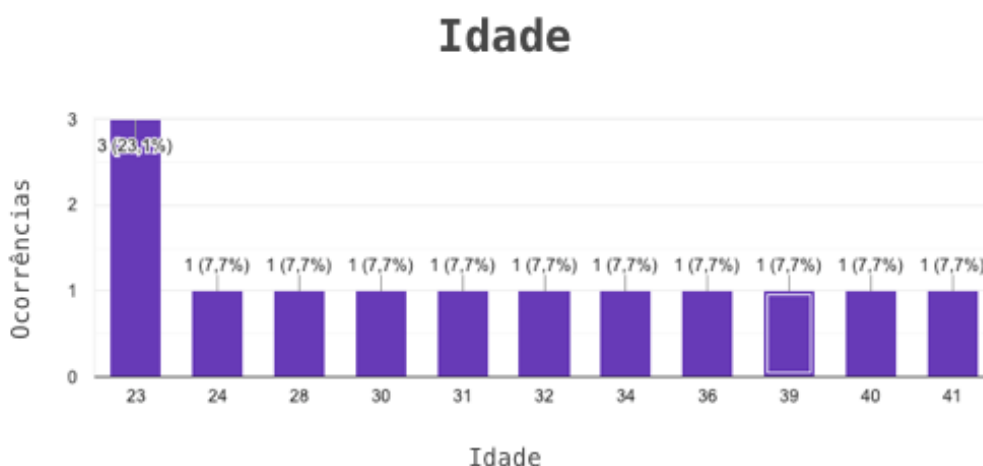


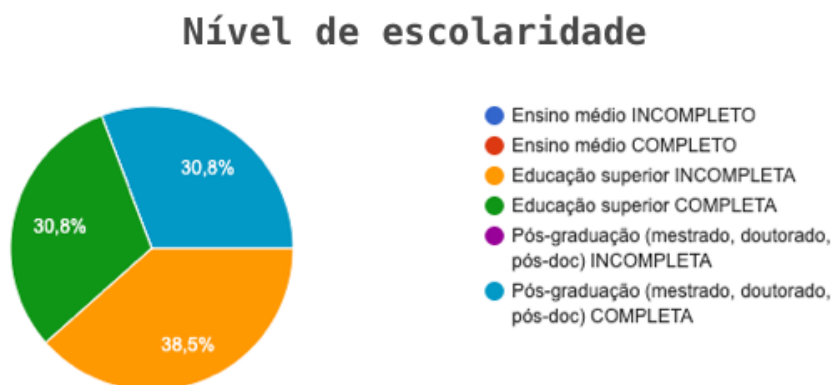
Figura 7.2: Gráfico das idades dos participantes da avaliação com usuários



No que diz respeito ao nível de escolaridade, as respostas foram as mais variadas,

como mostra a Figura 7.3. É possível notar que todas as pessoas que responderam ao questionário possuem ao menos nível superior incompleto de ensino. Esta informação não foi uma surpresa para o autor, dado que os testes foram realizados, principalmente, em ambiente acadêmico.

Figura 7.3: Gráfico do nível de escolaridade dos participantes da avaliação com usuários



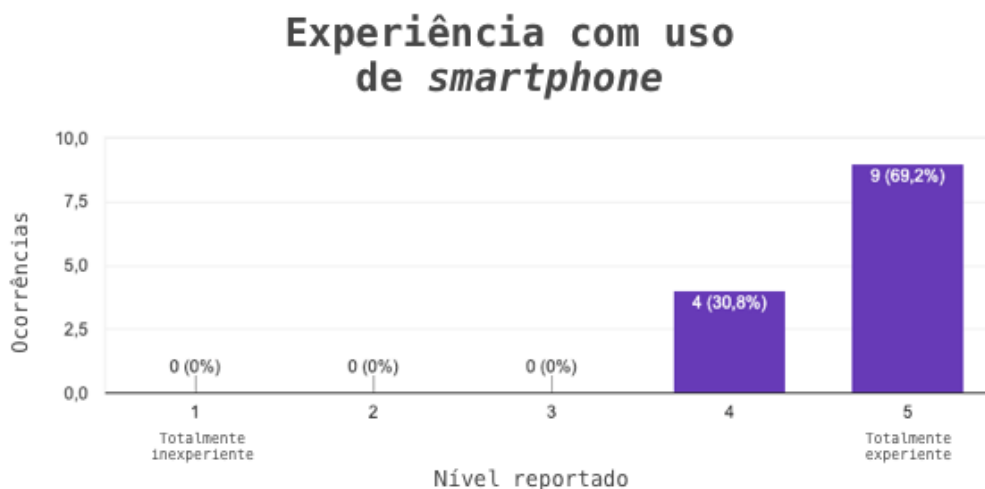
Na auto-avaliação feita pelo usuário com relação ao seu nível de experiência com o uso de internet, a grande maioria se considerou totalmente experiente, como mostra a Figura 7.4. Nenhuma das respostas foi abaixo de 4 na escala numérica de 1 a 5. Já na questão da experiência com o uso de *smartphone*, os participantes ficaram mais divididos, como mostra a Figura 7.5, mas novamente, não houve respostas com valor numérico abaixo de 4.

Figura 7.4: Gráfico do nível de experiência dos participantes com o uso da internet



Por fim, a Figura 7.6 apresenta os resultados da pergunta de múltipla escolha que visava descobrir os aplicativos relacionadas com mobilidade urbana que os participantes

Figura 7.5: Gráfico do nível de experiência dos participantes com o uso de *smartphone*



utilizam. É visível que aplicativos de transporte pago, como Uber, 99 e Cabify, são os mais utilizados. O Uber, por exemplo, foi apontado em 100% das respostas dos participantes. Além destes, destacam-se o Waze, bastante utilizado por pessoas que dirigem no seu dia-a-dia, e Moovit, aplicativo que traz informações das linhas de ônibus da cidade. Aplicativos de caronas, como Waze Carpool e BlaBlaCar, tiveram presença quase nula nas respostas. Este teve 1 menção apenas, enquanto que aquele não teve uma sequer.

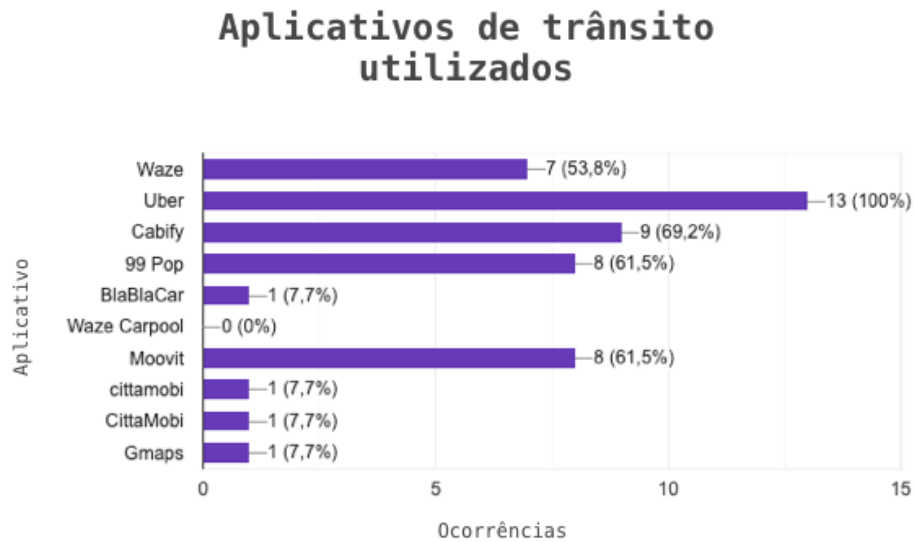
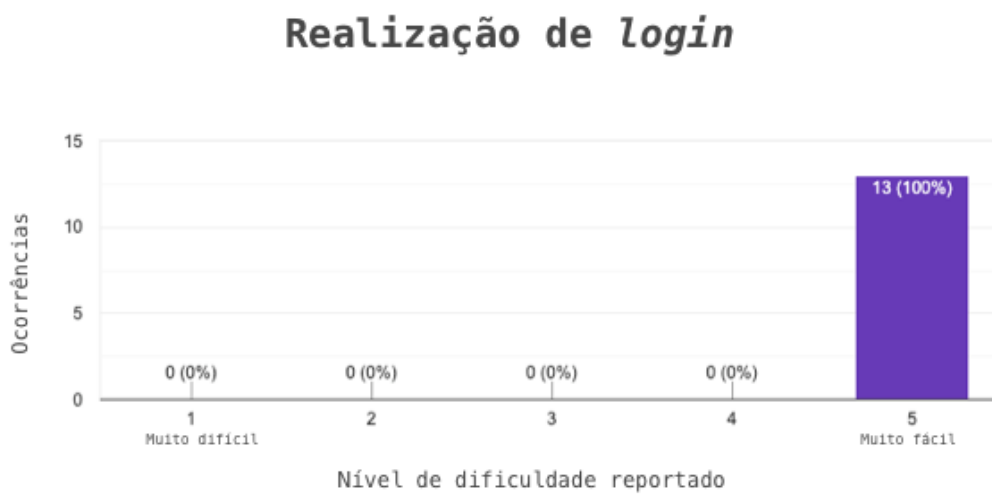
A ausência de serviços de corridas compartilhadas dentre os mais citados pelos participantes tem origem, na opinião do autor, da falta de incentivo de uso para o usuário final. A falta de conhecimento dos benefícios da prática de caronas compartilhadas, junto com o receio que este tipo de prática ainda causa, e o seu surgimento relativamente recente, faz com que a adesão do público ainda não seja expressiva.

7.3.2 Análise do Teste de Usabilidade da Etapa Inicial

Ao longo dos testes, foi solicitado aos usuários que executassem tarefas relacionadas às funcionalidades disponíveis no UFRGS Caronas. Nesta subseção, são apresentadas e analisadas as avaliações dos participantes sobre cada *feature* inclusa na primeira parte dos testes, que englobava todos os usuários.

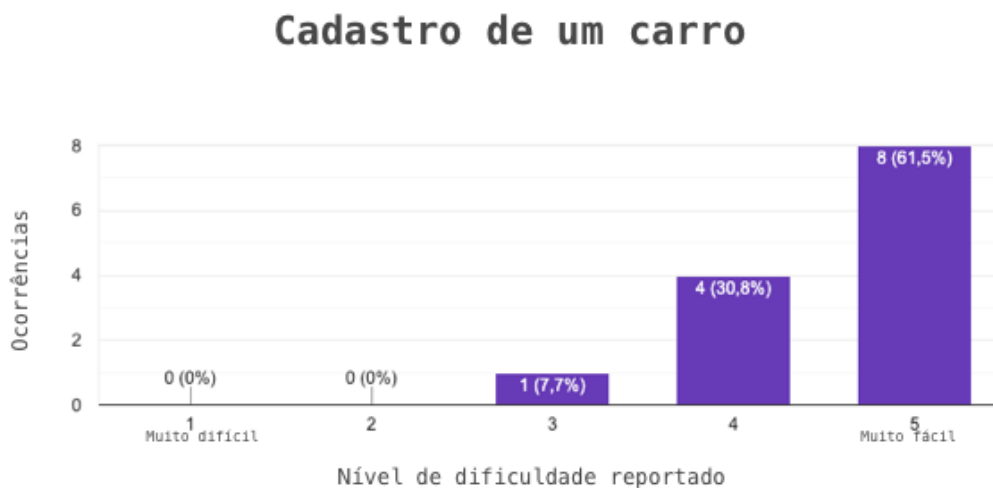
A primeira tarefa avaliada foi a realização de *login* no aplicativo utilizando as credenciais da universidade. Todos os participantes avaliaram a tarefa como “Muito fácil”, como mostrado na Figura 7.7. Este resultado não surpreendeu o autor, visto que a tarefa é deveras simples.

Figura 7.6: Aplicativos utilizados pelos participantes

Figura 7.7: Avaliação da realização de *login*

Em seguida, foi avaliada o cadastro de um carro do usuário. As respostas foram majoritariamente positivas, mas houve uma avaliação neutra (3 na escala de 1 a 5). As demais ficaram divididas entre o nível 4, com 4 respostas, e 5 (que corresponde a “Muito Fácil”), com 8 respostas. A Figura 7.8 mostra um gráfico com as respostas desta tarefa.

Figura 7.8: Avaliação do cadastro de um carro



A próxima tarefa a ser executada pelos participantes foi o cadastro de um oferecimento de carona. Esta tarefa foi a mais longa de todas (considerando tarefas realizadas por toda a amostra de participantes), pois o fluxo de cadastro de novo oferecimento de carona foi dividido em três telas. As avaliações foram bastante positivas: 10 respostas (em torno de 75%) consideraram a tarefa “Muito fácil” (nota 5), e as demais avaliações obtiveram nota 4 de 5, como mostra a Figura 7.9. O autor acredita que as boas avaliações se devem justamente pelo fluxo ter sido dividido em seções que coletam tipos diferentes de informações, pois isto evita confusão e fadiga por parte do usuário, que possui um tipo de informação no qual focar por vez.

Após criar um oferecimento de carona, o usuário tinha a tarefa de buscar por caronas oferecidas por outros usuários, utilizando-se de filtros para refinar a busca. O filtro escolhido foi o local de origem da carona sendo o Campus Saúde. Novamente, os resultados foram positivos, como mostra a Figura 7.10. Cerca de 70% dos participantes consideraram a tarefa “Muito fácil”, e deram nota 5. Todos os demais avaliaram a tarefa com nota 4.

Por fim, complementando a tarefa anterior, foi solicitado aos participantes que criassem um pedido de carona para alguma das caronas resultantes da busca realizada. Nesta tarefa, houve uma avaliação neutra, com nota 3, como pode ser visto na Figura

Figura 7.9: Avaliação do oferecimento de uma carona

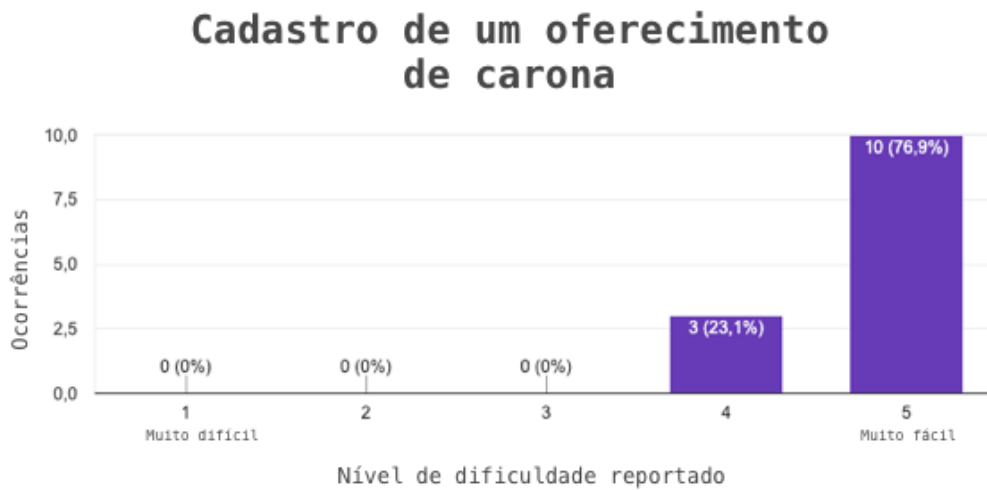
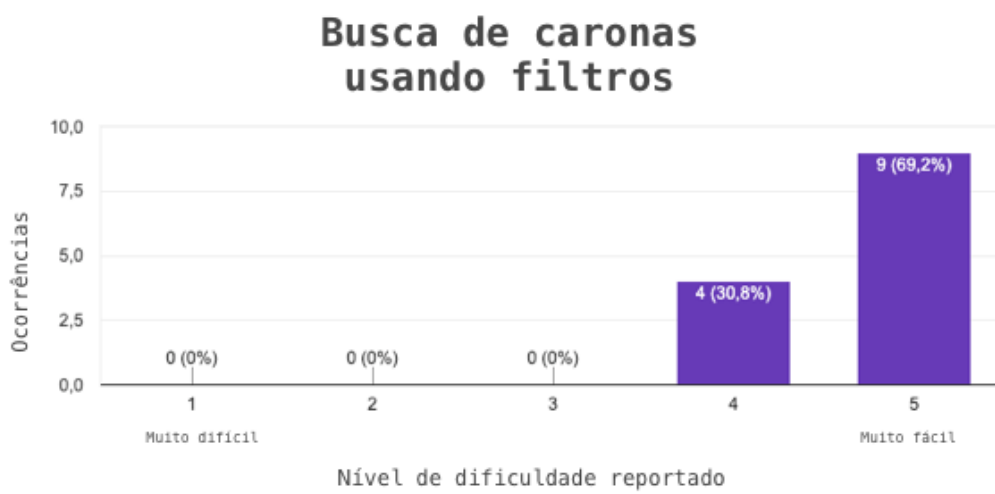
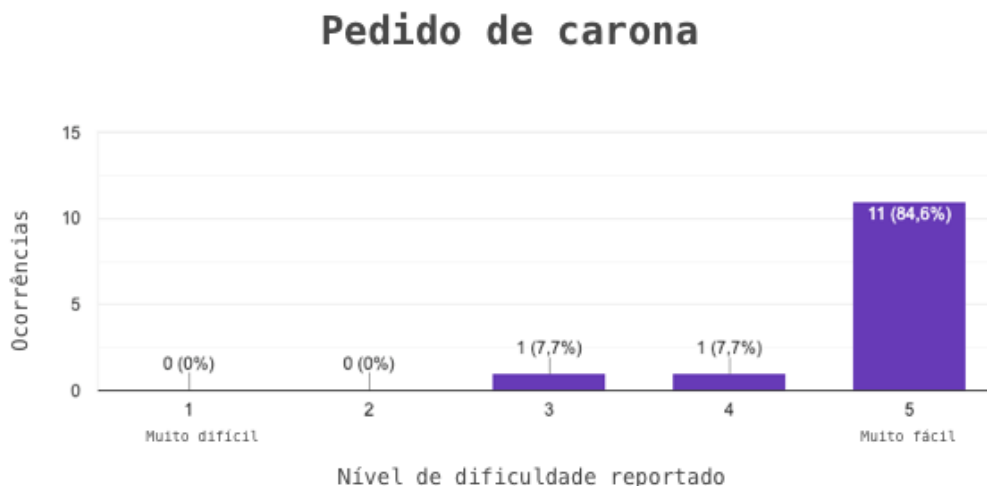


Figura 7.10: Avaliação da busca por caronas com filtro



7.11. Das demais respostas, uma avaliou a tarefa com nota 4, e as demais, com nota 5.

Figura 7.11: Avaliação da criação de um pedido de carona



7.3.3 Análise do Teste de Usabilidade com mulheres

Após as tarefas obrigatórias para todos os usuários, a avaliação sofria uma desambiguação, onde o gênero da pessoa participante definia o caminho a ser tomado. Caso fosse do gênero feminino, a pessoa teria duas tarefas extras para avaliar. Caso contrário, iria direto à fase final. Esta subseção está dedicada à apresentação e análise dos resultados das tarefas testadas especialmente com as 2 usuárias mulheres que participaram dos testes.

A primeira tarefa consistia em cadastrar um oferecimento de carona apenas para mulheres. Das respostas obtidas, 1 avaliou a tarefa com nota 4, e a outra com nota 5, como pode ser observado na Figura 7.12

Em seguida, as usuárias tinham a tarefa de buscar por caronas oferecidas por outras pessoas, utilizando o filtro (disponível apenas para pessoas do gênero feminino) que fazia com que fossem exibidas apenas caronas oferecidas por mulheres. Os resultados foram os mesmos em todas as respostas: nota 4 de 5, como mostra a Figura 7.13.

O autor entende que a amostragem de participantes nesta etapa foi pequena. Porém, as respostas mantiveram-se consistentes com o restante dos testes aplicados.

Figura 7.12: Avaliação do oferecimento de carona apenas para mulheres

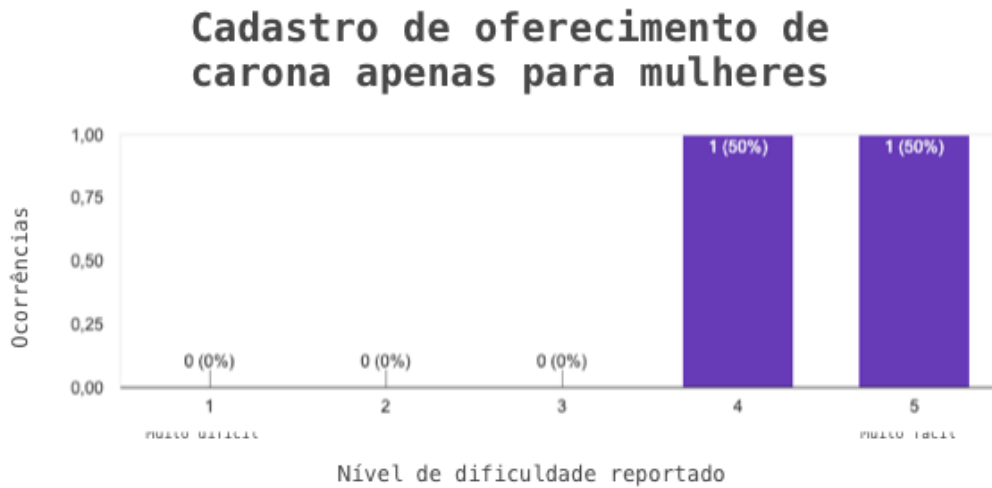
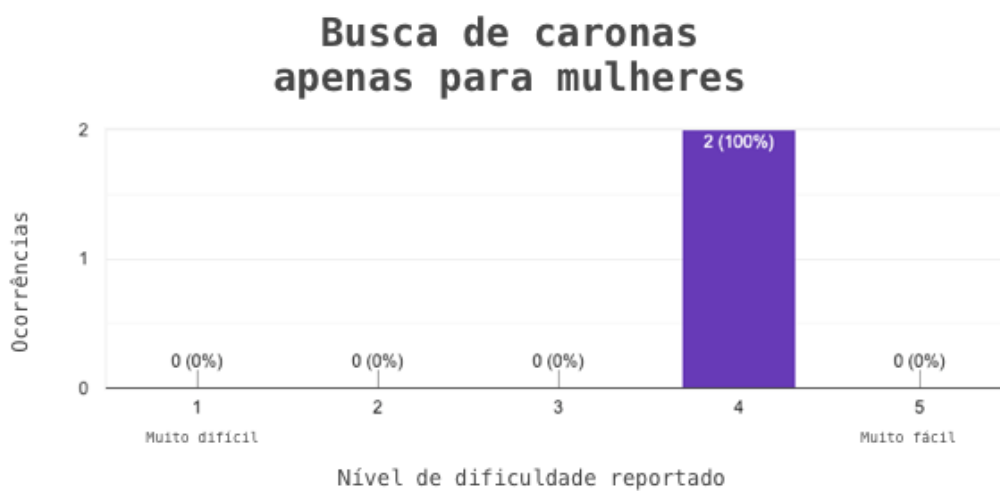


Figura 7.13: Avaliação da busca de caronas apenas para mulheres

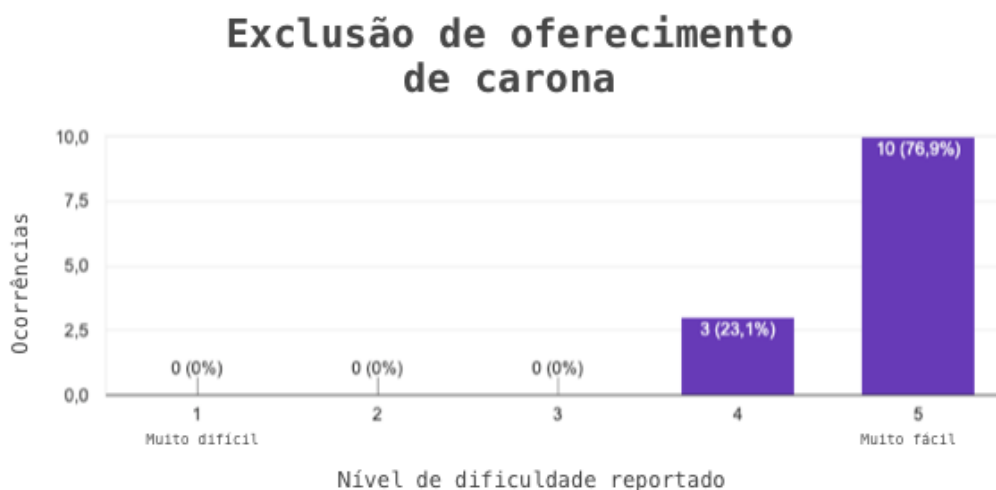


7.3.4 Análise do Teste de Usabilidade da Etapa Final

Após a etapa intermediária para participantes do gênero feminino, e após a etapa inicial para os demais, as pessoas participantes adentravam na etapa final dos testes, que consiste em duas tarefas.

Primeiramente, o participante era instruído a excluir o oferecimento de carona que havia criado anteriormente. Esta tarefa obteve resultados bastante positivos: aproximadamente 75% das respostas avaliaram a tarefa como “Muito fácil”, com nota 5. As demais avaliações foram nota 4 de 5, como é possível ver na Figura 7.14

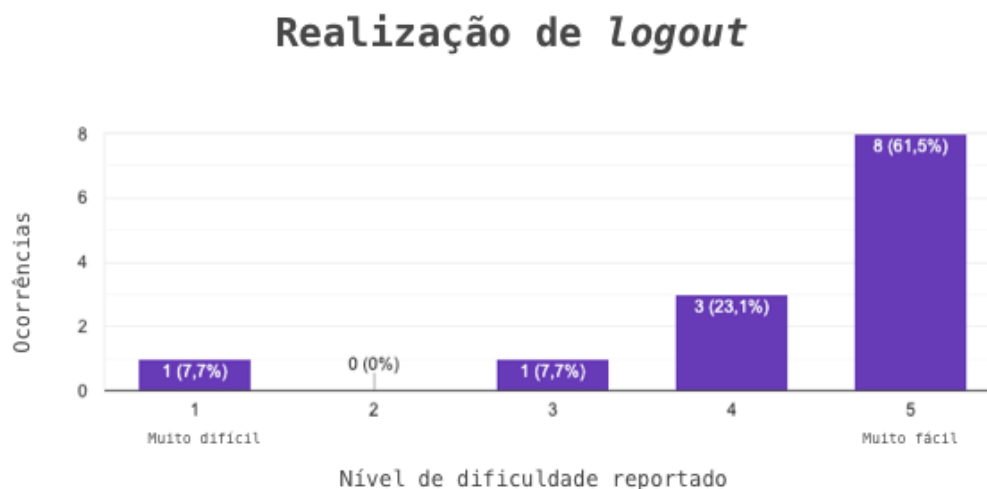
Figura 7.14: Avaliação da tarefa de excluir um oferecimento de carona



Por fim, era solicitado ao usuário que realizasse *logout* do aplicativo. Esta tarefa, apesar de bastante simples, obteve algumas avaliações diferenciadas, como mostra a Figura 7.15. Cerca de 60% dos participantes avaliou a tarefa com nota 5 em facilidade, e 23% avaliaram com nota 4. Nas demais avaliações, houve uma resposta neutra, com nota 3, e uma resposta com nota mínima (1 na escala de 1 a 5) para a facilidade de realizar a tarefa. O motivo destas avaliações neutra/negativas será discutido na sessão seguinte, onde são apresentadas as sugestões e observações das pessoas que participaram dos testes.

7.3.5 Sugestões

Ao final do formulário de avaliação do UFRGS Caronas, o participante podia inserir um comentário sobre sua experiência com o aplicativo. Esta subseção é dedicada a

Figura 7.15: Avaliação da tarefa de fazer *logout* do aplicativo

apresentar as principais observações e sugestões colhidas nas respostas dos participantes.

Um ponto que apareceu em mais de uma resposta foi a dificuldade de realizar *logout* do aplicativo. Em duas ocasiões, foi criticada a representação visual do botão onde o usuário deve clicar para deslogar-se: em uma delas, foi citado o tamanho do botão, que seria demasiado pequeno; em outra, foi sugerido que a cor do botão fosse mais vibrante, que desse mais destaque para esta opção. Uma terceira observação foi feita a respeito da falta de uma confirmação com o usuário, através de uma caixa de diálogo padrão do sistema. Isto evitaria que o usuário se deslogasse do aplicativo ao tocar no botão de *logout* acidentalmente.

Foi citada também, a expectativa de comportamentos que são padrão no iOS e até em outros aplicativos, como Gmail¹, como o gesto de arrastar um item de uma lista para o lado para excluí-lo. Por outro lado, foi observado certo desconforto com um comportamento em específico que é padrão do sistema: ao trocar de uma aba para outra, na parte inferior do aplicativo, é salvo o estado da aba o usuário estava. Isto ocasionou uma situação em que, com o objetivo de acessar a tela de perfil, o usuário se deparasse com a tela de gerência de carros, que havia sido aberta previamente.

Após o cumprimento de uma tarefa como cadastro de um oferecimento de carona e pedido de carona, é mostrada ao usuário uma tela lúdica com uma ilustração e um *label* de texto dizendo que a tarefa foi executada com sucesso. Após, o usuário deve fechar esta tela através de um botão no canto esquerdo superior (com o formato de um “X”) para voltar ao fluxo normal do aplicativo. Um usuário comentou a respeito da falta de praticidade de

¹<<https://apps.apple.com/us/app/gmail-email-by-google/id422689480>> Acesso em outubro de 2019

se ter apenas este botão longe do alcance do usuário para voltar. Foi sugerido um botão de fechar na parte inferior da tela, mais fácil de alcançar para o usuário.

Com as observações colhidas, o autor conclui que algumas alterações poderiam ser feitas com o objetivo de melhorar a experiência do usuário ao utilizar o UFRGS Caronas. Algumas delas são:

- Alteração do *logout*: aumentar o tamanho e alterar a cor do botão, no que se refere à parte visual, e acrescentar uma confirmação interrogando se o usuário deseja, de fato, realizar *logout* do aplicativo;
- Analisar a possibilidade de implementação e/ou alteração dos comportamentos padrão do iOS de acordo com o caso: alguns comportamentos que são padrão do sistema operacional do iPhone causaram certo estranhamento, enquanto que outros, ausentes no UFRGS Caronas, fizeram falta. Um trabalho futuro por parte do autor é analisar onde faz sentido manter os comportamentos padrão do sistema, e onde é necessários adicionar comportamentos padrão que não estão implementados;
- Criar formas mais práticas de sair da tela de “Sucesso”: o autor chegou a conclusão de que seria útil implementar um mecanismo alternativo para que o usuário feche a tela que indica sucesso no cumprimento de uma tarefa. A solução encontrada foi capturar o toque do usuário em qualquer posição da tela, e então fechá-la, sem a necessidade de o toque ter que ser, obrigatoriamente, no botão de fechar.

8 CONCLUSÕES

Neste trabalho, foram implementadas as principais funcionalidades definidas no projeto UFRGS Caronas, idealizado no CPD da UFRGS com o objetivo de prover à comunidade da universidade uma ferramenta para facilitar a troca de caronas voluntárias. Estas funcionalidades, como estão implementadas no protótipo resultante deste trabalho, foram descritas em detalhes no Capítulo 6.

O protótipo resultante possui pontos em comum com outras ferramentas similares existentes no mercado, como BlaBlaCar, Waze Carpool e Zumpy. Um ponto que merece destaque é a presença de funcionalidades que incitam maior segurança para usuárias do gênero feminino, através da possibilidade de oferecer caronas apenas para outras usuárias mulheres, e de poder buscar por caronas oferecidas apenas por outras mulheres. Além disso, o UFRGS Caronas possui um grande diferencial em relação aos aplicativos citados anteriormente: o *login* com as credenciais da universidade. Por ter como público alvo a comunidade da UFRGS, foi implementado um mecanismo de *login* similar ao de outros aplicativos da universidade, como o UFRGS Mobile¹, onde o usuário insere o número de seu cartão da UFRGS e sua senha do portal da universidade. O autor acredita que isto traz maior sensação de segurança para os usuários, uma vez que o acesso é restrito a pessoas da comunidade da UFRGS.

O desenvolvimento deste trabalho culminou em um protótipo funcional do projeto UFRGS Caronas para a versão iOS. Por questões de tempo, ao longo do desenvolvimento, houve limitações no escopo do trabalho. Algumas funcionalidades que haviam sido planejadas inicialmente acabaram sendo postergadas para um momento futuro. São elas:

- **Edição dos dados do usuário:** consiste em permitir que o usuário altere seus dados pessoais, como sua foto do perfil e número de telefone, através do aplicativo;
- **Oferecimento de caronas sucessivas:** permitir que um oferecimento de carona seja repetido a cada semana, no mesmo dia da semana. Isto faria com que um usuário motorista não precisasse recadastrar a mesma carona a cada semana, caso queira oferecê-la mais de uma vez.
- **Busca de caronas por dia da semana:** complementando a funcionalidade acima, esta faz com que o usuário possa buscar por caronas selecionando o dia da semana em que ela irá ocorrer.
- **Busca de caronas por localização no mapa:** permitir que o usuário busque por ca-

¹<<https://apps.apple.com/br/app/ufrgs-mobile/id1058897658>> Acesso em outubro de 2019

ronas que tenham como origem ou destino um local que não seja um campus (opção “Outro” na versão atual do UFRGS Caronas) selecionando sua localização atual em um mapa no aplicativo. A API, então, retorna uma lista de caronas cuja origem ou destino (dependendo da escolha do usuário) fique em um raio de x quilômetros da localização escolhida pelo usuário, onde x é um número a ser definido;

- **Avaliação do motorista e do caronista:** permitir que os usuários envolvidos na carona dessem uma nota referente à outra pessoa: no caso de quem recebeu a carona, avaliaria o motorista, e vice-versa. Isto traria maior controle sobre eventuais irregularidades providas de mau comportamento dos usuários do sistema.
- **Pedido de carona em grupo:** permitir que usuários peçam caronas reservando mais de um lugar no carro, quando disponível. Isto possibilitaria que uma pessoa apenas pedisse carona para um grupo (com um tamanho máximo a ser definido).

Além das funcionalidades citadas acima, o autor vê como trabalho futuro para o projeto UFRGS Caronas a implementação da versão Android do aplicativo. Com isto feito, poderia-se avaliar a possibilidade da implantação do aplicativo no mundo real, trazendo benefícios reais para a comunidade.

Apesar das limitações encontradas, o autor avalia o protótipo resultante deste trabalho como satisfatório, visto que as principais funcionalidades foram implementadas e os testes feitos com usuários mostram que a usabilidade do UFRGS Caronas alcançou um nível adequado, apesar de observações negativas pontualmente realizadas pelos participantes dos testes, que certamente contribuirão para a melhora da usabilidade do aplicativo em um momento futuro. O autor acredita que após a implementação de algumas das melhorias citadas acima (assim como a implementação da versão Android do aplicativo), o UFRGS Caronas poderia ser lançado e enfim, ser utilizado de fato, pela comunidade da UFRGS.

Este trabalho envolveu, além do desenvolvimento de *software*, uma pesquisa acerca do contexto em que o aplicativo final poderia vir a ser utilizado no mundo real, como pode ser visto na seção sobre Mobilidade Urbana no Capítulo 2. Isto forneceu ao autor uma visão mais ampla da situação atual da mobilidade urbana na conjuntura atual do país.

Com os estudos realizados a respeito de mobilidade urbana, o autor chegou à conclusão de que uma ferramenta que facilite a troca de caronas universitárias poderia sim impactar positivamente a percepção da comunidade acadêmica sobre o deslocamento até os campi da UFRGS. O autor não acredita que o impacto seria grande o suficiente para que ocorresse uma mudança drástica na mobilidade urbana da cidade, mas que traria benefí-

cios pontuais a pessoas que o utilizassem, e assim o objetivo do projeto estaria cumprido.

REFERÊNCIAS

- Altexsoft. **Agile Project Management: Best Practices and Methodologies**. 2016. <<https://www.altexsoft.com/media/2016/04/Agile-Project-Management-Best-Practices-and-Methodologies-Whitepaper.pdf>>.
- Apple Developer. **Cocoa Core Components**. 2018. <<https://developer.apple.com/library/archive/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html>>.
- ARASAKI, P. K. et al. Contribuições da carona solidária no processo de compartilhamento do conhecimento. **Navus - Revista de Gestão e Tecnologia**, Florianópolis, SC, Brasil, v. 6, 2016. ISSN 2237-4558. Disponível em: <<https://www.redalyc.org/pdf/3504/350454047004.pdf>>.
- BERSON, A. **Client/Server Architecture (J. Ranade Series on Computer Communications)**. McGraw-Hill (Tx), 1992. ISBN 0070050767. Disponível em: <<https://www.xarg.org/ref/a/0070050767/>>.
- BlaBlaCar. **BlaBlaCar atinge marca de 5 milhões de usuários no Brasil**. 2019. <<https://blog.blablacar.com.br/newsroom/noticias/blablacar-atinge-marca-de-5-milhoes-de-usuarios-no-brasil>>.
- CARVALHO, C. H. R. de. Mobilidade urbana sustentável: Conceitos, tendências e reflexões. **Texto para Discussão**, Ipea, Brasília, DF, Brasil, v. 2194, maio 2016. ISSN 1415-4765. Disponível em: <http://repositorio.ipea.gov.br/bitstream/11058/6637/1/td_2194.pdf>.
- CARÚS, A. **Dia Nacional do Trânsito: frota de veículos cresce 70dez anos**. 2019. <<http://www.camarapoa.rs.gov.br/noticias/dia-nacional-do-transito-frota-de-veiculos-cresce-70-em-dez-anos>>.
- CINTRA, M. A crise do trânsito em São Paulo e seus custos. **Navus - Revista de Gestão e Tecnologia**, São Paulo, SP, Brasil, v. 12, 2013. ISSN 1806-8979. Disponível em: <<http://bibliotecadigital.fgv.br/ojs/index.php/gvexecutivo/article/view/20728/19462>>.
- CONCEICAO, L. dos S. **Web service para acesso a dados da aplicação caronas**. 2017. <<https://www.lume.ufrgs.br/handle/10183/169003>>.
- DEACON, J. Model-view-controller (mvc) architecture. august 1995. Disponível em: <<http://www.rareparts.com/pdf/MVC.pdf>>.
- FOSTAINI, R. P. **Entendendo o Repository Pattern**. 2018. <<https://medium.com/@renicius.pagotto/entendendo-o-repository-pattern-fcdd0c36b63b>>.
- GNIPPER, P. **Quais são os apps de transporte que permitem escolher uma motorista mulher?** 2019. <<https://canaltech.com.br/apps/quais-sao-os-apps-de-transporte-que-permitem-escolher-uma-motorista-mulher>>.
- GOMES, P. **Brasil registra mais de 180 estupros por dia; número é o maior desde 2009**. 2019. <<https://www1.folha.uol.com.br/cotidiano/2019/09/brasil-registra-mais-de-180-estupros-por-dia-numero-e-o-maior-desde-2009.shtml>>.

- Google. **Google Maps and Waze, outsmarting traffic together**. 2013. <<https://googleblog.blogspot.com/2013/06/google-maps-and-waze-outsmarting.html>>.
- HAWKINS, A. J. **Google's Waze jumps on the carpool bandwagon with new Bay Area pilot**. 2016. <<https://www.theverge.com/2016/5/16/11685396/google-waze-carpool-pilot-san-francisco-uber-lyft>>.
- LIKERT, R. **A technique for the measurement of attitudes**. The Science Press, 1932. Disponível em: <https://books.google.com.br/books/about/A_Technique_for_the_Measurement_of_Attit.html?id=9rotAAAAYAAJ>.
- Lokesh Gupta. **Java Singleton Pattern Explained**. 2019. <<https://howtodoinjava.com/design-patterns/creational/singleton-design-pattern-in-java>>.
- NEWMAN, S. **Pattern: Backends For Frontends**. 2015. <<https://samnewman.io/patterns/architectural/bff/>>.
- O Globo. **Waze Carpool chega ao Brasil com caronas a partir de R\$ 2**. 2018. <<https://oglobo.globo.com/economia/waze-carpool-chega-ao-brasil-com-caronas-partir-de-2-22997582>>.
- PERES, R.; LASO-MARSETTI, F. **Model-View-Controller (MVC) in iOS – A Modern Approach**. 2019. <<https://www.raywenderlich.com/1000705-model-view-controller-mvc-in-ios-a-modern-approach>>.
- Polcode Team. **Trello in Software Development—How to Manage Complex Projects**. 2018. <<https://polcode.com/blog/trello-in-software-development-how-to-manage-complex-projects>>.
- PRESSMAN, R. S. **Engenharia de Software**. Pearson, 1995. ISBN 8534602379. Disponível em: <<https://www.xarg.org/ref/a/8534602379/>>.
- SILVA, E. F. **Meio ambiente mobilidade urbana**. São Paulo: Editora SENAC São Paulo, 2017. ISBN 9788539608065. Disponível em: <https://books.google.com.br/books/about/Meio_ambiente_mobilidade_urbana.html?id=IBisDwAAQBAJ>.
- SKERRETT, I. **Eclipse Community Survey 2014 Results**. 2014. <<https://dzone.com/articles/eclipse-community-survey-2014>>.
- SOMMERVILLE, I. **Engenharia de software**. PEARSON BRASIL, 2011. ISBN 9788579361081. Disponível em: <<https://books.google.com.br/books?id=H4u5ygAACAAJ>>.
- Stack Overflow. **Developer Survey Results 2018**. 2018. <<https://insights.stackoverflow.com/survey/2018>>.
- SWIFT.ORG. **About Swift**. 2019. <<https://swift.org/about>>.
- The Manifest. **How to Publish Your App on Apple's App Store in 2018**. 2018. <https://medium.com/@the_manifest/how-to-publish-your-app-on-apples-app-store-in-2018-f76f22a5c33a>.
- Waze for Brands. **Saiba mais sobre o Waze**. 2019. <<https://support.google.com/wazeforbrands/answer/7496007>>.

Zumpy. **Descubra como ter maiores ganhos e previsibilidade para seus dias!** 2019.
<<https://www.zumpy.com.br/motorista>>.

APÊNDICE A — FORMULÁRIO PRÉ-TESTE

Experimento com usuários

Este experimento tem o objetivo de avaliar a experiência de usuário do aplicativo UFRGS Caronas para a plataforma iOS. O experimento todo leva cerca de 15 minutos, e consiste nos seguintes procedimentos:

1. Responder um formulário pré-teste, com o objetivo de entender melhor o perfil das pessoas que aceitaram participar da avaliação com usuários;
2. Receber instrução acerca do uso do aplicativo;
3. Executar uma sequência de tarefas pré-estabelecidas;
4. Após cada grupo de tarefas, responder questões avaliando sua experiência.

Os participantes podem a qualquer momento e, por qualquer motivo particular, interromper os testes. Os dados obtidos ao longo do experimento serão utilizados apenas neste estudo e de forma totalmente anônima.

Sobre os riscos: algumas pessoas podem apresentar cansaço em algum grau durante realização do experimento.

*Obrigatório

1. Caso esteja de acordo, marque a opção abaixo *

Marcar apenas uma oval.

Aceito participar deste experimento. Declaro que fui devidamente informado sobre os objetivos da pesquisa, os procedimentos envolvidos nos testes aos quais vou me submeter e os possíveis riscos decorrentes de minha participação. Foi-me garantido o sigilo de minhas informações e o direito de retirar minha participação a qualquer momento.

Não aceito participar. *Pare de preencher este formulário.*

Formulário pré-teste

2. Gênero *

Marcar apenas uma oval.

Feminino

Masculino

Prefiro não dizer

Outro: _____

3. Idade *

4. Nível de escolaridade *

Marcar apenas uma oval.

Ensino médio INCOMPLETO

Ensino médio COMPLETO

Educação superior INCOMPLETA

Educação superior COMPLETA

Pós-graduação (mestrado, doutorado, pós-doc) INCOMPLETA

Pós-graduação (mestrado, doutorado, pós-doc) COMPLETA

Outro: _____

Tecnologias

Nesta seção, as perguntas buscam entender o seu grau de conhecimento prévio a respeito do uso de tecnologia. Marque nas respostas o quanto você sente que domina a habilidade descrita na pergunta.

5. Experiência com uso da internet *

Marcar apenas uma oval.

	1	2	3	4	5	
Totalmente inexperiente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Totalmente experiente

6. Experiência com smartphone *

Marcar apenas uma oval.

	1	2	3	4	5	
Totalmente inexperiente	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Totalmente experiente

7. Selecione abaixo os aplicativos de trânsito/transporte que você utiliza

Marque todas que se aplicam.

- Waze
- Uber
- Cabify
- 99 Pop
- BlaBlaCar
- Waze Carpool
- Moovit
- Outro: _____

APÊNDICE B — FORMULÁRIO PÓS PRIMEIRA ETAPA

Etapa inicial

Avalie sua experiência ao realizar as tarefas desta etapa, com notas de 1 a 5. Em cada questão está explicitado o que cada valor significa.

***Obrigatório**

- 1. Na sua opinião, qual foi o nível de dificuldade de realizar login no aplicativo com as credenciais da universidade? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

- 2. Na sua opinião, qual foi o nível de dificuldade de cadastrar um carro no aplicativo? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

- 3. Na sua opinião, qual foi o nível de dificuldade de cadastrar um oferecimento de carona? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

- 4. Na sua opinião, qual foi o nível de dificuldade de buscar por caronas saindo do Campus Saúde? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

- 5. Na sua opinião, qual foi o nível de dificuldade de fazer um pedido de carona? ***

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

APÊNDICE C — FORMULÁRIO PÓS ETAPA APENAS COM MULHERES

Etapa apenas para mulheres

Avalie sua experiência ao realizar as tarefas desta etapa, com notas de 1 a 5. Em cada questão está explicitado o que cada valor significa.

***Obrigatório**

1. Na sua opinião, qual foi o nível de dificuldade de cadastrar um oferecimento de carona apenas para mulheres? *

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

2. Na sua opinião, qual foi o nível de dificuldade de buscar por caronas apenas para mulheres? *

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

Powered by



APÊNDICE D — FORMULÁRIO PÓS ETAPA FINAL

Etapa Final

Avalie sua experiência ao realizar as tarefas desta etapa, com notas de 1 a 5. Em cada questão está explicitado o que cada valor significa.

***Obrigatório**

1. Na sua opinião, qual foi o nível de dificuldade de excluir um oferecimento de carona? *

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

2. Na sua opinião, qual foi o nível de dificuldade de deslogar-se do aplicativo? *

Marcar apenas uma oval.

	1	2	3	4	5	
Muito difícil	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito fácil

3. De forma geral, como você avalia seu desempenho executando as atividades na plataforma? *

Marcar apenas uma oval.

	1	2	3	4	5	
Muito ruim	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito bom

4. Descreva brevemente as principais dificuldades que você encontrou durante a realização de alguma atividade na plataforma, se houve alguma.

Powered by

