

# Colabore: Uma Aplicação para os Interessados em Compartilhar

Amaury Alexandrino Costa, Guilherme Sanzio Menezes Zallio, Álisson Rabelo Arantes

PUC Minas em Betim  
Bacharelado em Sistemas de Informação

amaury.costa@sga.pucminas.br, guilherme.zallio@sga.pucminas.br,  
alissonr@pucminas.br

## 1. Introdução

Estar conectado hoje em dia virou algo trivial. Percebe-se uma invasão massiva das tecnologias de informação nos ambientes de convivência, ao estarem presentes em grande parte das atividades. Com isso, a comunicação se estabelece de forma mais rápida e fácil, ao aproximar pessoas e informações. Isso é importante, pois ter a informação em mãos de uma forma consistente e rápida, e se possível em tempo real e de qualquer lugar que se esteja, é extremamente relevante para a assertividade nas tomadas de decisão, sejam elas a nível corporativo ou não. Juntamente a tudo isso, está em crescimento uma forma alternativa de fazer negócios. Segundo o SEBRAE (2017), diante de problemas sociais e ambientais que se agravam, cada vez mais o acúmulo é substituído pelo compartilhamento, onde ganha força o conceito de economia colaborativa, ou compartilhada, em que bens e serviços podem ser compartilhados entre os interessados, sem necessariamente haver algum intermediário, de forma a reduzir despesas. Esse conceito é fomentado pelo constante avanço tecnológico, de forma que já é crescente a utilização de serviços *online*, como páginas *Web* ou aplicativos, para essa finalidade. A economia colaborativa ganha força, e já são oferecidos serviços voltados para as áreas de transporte e mobilidade urbana, que são o foco do trabalho (SEBRAE 2017).

Ao tomar como base as áreas de transporte e mobilidade urbana no Brasil, percebe-se uma grande insatisfação da população para com o transporte público atualmente. As maiores queixas são relativas a problemas com a infraestrutura e serviço prestado, com a insegurança em terminais e pontos de parada, além de altas tarifas dos meios de transporte coletivo urbano, ao considerar a média salarial dos brasileiros. Além disso, com um número cada vez maior de carros particulares nas ruas, pode-se destacar a superlotação das vias públicas, o que também leva a problemas de mobilidade. A partir disso, como pode-se atrelar o conceito de economia colaborativa com o uso das tecnologias de informação, de forma a contribuir com a mobilidade urbana no Brasil?

A mobilidade urbana é um grande problema nas grandes cidades do mundo. No cenário brasileiro atual, em instituições de ensino como universidades, por exemplo, observa-se que muitos estudantes, principalmente os que moram mais longe de onde estudam, sentem dificuldades com relação à mobilidade a esses locais, principalmente por problemas com o transporte público. Esse cenário pode se estender também a eventos e locais com grande aglomeração de pessoas, onde dependendo do horário se

torna difícil a chegada ou o retorno do local. Portanto, há uma procura muito grande por meios de transporte alternativos, como caronas, que possibilitam uma maior praticidade e em um menor custo possível. A partir desse contexto, aliado ao desenvolvimento tecnológico que se tem atualmente, percebe-se que há uma demanda por um aplicativo, que através de um sistema de notificações, pudesse aproximar pessoas que tivessem um destino comum, e que estivessem dispostas a negociar uma carona.

O objetivo deste trabalho é desenvolver um aplicativo para dispositivos móveis que ofereça uma plataforma simplificada de comunicação, de forma a aproximar pessoas que queiram negociar uma carona.

Este texto está estruturado em 5 seções. A Seção 2 apresenta o referencial que oferece uma sustentação teórica a respeito do assunto tratado. A seção 3 descreve os trabalhos relacionados. Na Seção 4 é apresentada a metodologia do trabalho, com o tipo, suas etapas e ambiente de desenvolvimento. Na Seção 5 são apresentados aspectos relativos ao desenvolvimento. Por último, é apresentada a conclusão do trabalho.

## **2. Referencial Teórico**

Nesta seção são descritos os principais conceitos, classificações e técnicas relacionados ao desenvolvimento de *software* e computação móvel.

### **2.1. Sistemas de Informação**

A área de conhecimento de sistemas de informação abrange o estudo da disponibilização da informação dentro das organizações, mediante a articulação da tecnologia com as necessidades de informação das diferentes áreas de negócio da empresa. Ou seja, um sistema de informação pode ser definido como um conjunto de componentes relacionados que coletam, processam, armazenam e distribuem informações para as tomadas de decisão e para o controle de uma organização (Audy 2011), (Laudon e Laudon 2011).

### **2.2. Projeto de Software**

Um projeto é um esforço temporário empreendido para criar um produto, serviço ou resultado exclusivo. O projeto de *software* é uma atividade criativa em que se identificam os componentes de *software* e seus relacionamentos com base nos requisitos do cliente. Em outras palavras, o projeto está na cabeça do programador, ou esboçado em um quadro ou em um papel. Ele trata de como resolver um problema, onde são definidos alguns processos para a construção do *software* (PMI 2013), (Sommerville 2011).

### **2.3. Engenharia de Software**

Durante as primeiras eras do desenvolvimento de *softwares*, por volta da década de 70, não se tinha muito a ideia de planejamento e análise de requisitos de *software*, onde os mesmos acabavam por serem produzidos de forma desorganizada, sem nenhum padrão ou documentação. Viu-se a necessidade, portanto, pelas dificuldades encontradas na época, de aplicar os conceitos de engenharia no desenvolvimento de *softwares*, tornando-o mais eficiente e organizado. Nessa época surgiu o termo engenharia de *software*, que é uma disciplina de engenharia cujo foco está em todos os aspectos da

produção de *software*, desde os estágios iniciais da especificação do sistema até sua manutenção, quando o sistema já está sendo usado (Pressman 2016), (Sommerville 2011).

O desenvolvimento de sistemas, de acordo com os princípios da engenharia de *software*, segue uma série de processos, que define um conjunto de atividades relacionadas que levam à produção de um produto de *software*, tais como especificação, projeto e implementação, validação, e evolução de *software* (Sommerville 2011).

Por volta da década de 90, um grande número de desenvolvedores de *softwares* propôs novos métodos ágeis de desenvolvimento, devido à insatisfação com as abordagens pesadas da engenharia de *software* nos anos passados. Estes novos métodos ágeis, também chamados de metodologias ágeis, permitiam que a equipe de desenvolvimento focasse no *software* em si, baseando-se em uma abordagem incremental para a especificação, o desenvolvimento e a entrega do sistema. Dentre as metodologias ágeis utilizadas atualmente, pode-se destacar o *Extreme Programming (XP)* e o *SCRUM* (Sommerville 2011).

Para auxiliar na descrição e no projeto de sistemas de *software*, pode-se utilizar a *UML (Unified Modeling Language)* que envolve a criação de diversos documentos, que podem ser textuais ou gráficos, denominados artefatos de *software*, e que são representados por meio de diagramas. Dentre os diagramas *UML*, destacam-se a modelagem de casos de uso (que é uma representação das funcionalidades externamente observáveis do sistema e dos elementos externos a ele e que interagem com o mesmo), e a modelagem de classes (que permite compreender como o sistema está estruturado internamente) (Fowler 2011), (Bezerra 2003).

## **2.4. Banco de Dados**

Dados são fatos conhecidos que possuem significado implícito, como por exemplo, nomes, números de telefone e endereços de pessoas conhecidas. Os dados podem ser relacionados e registrados em algum lugar, comumente chamado de banco de dados. Ou seja, genericamente, um banco de dados é uma coleção de dados relacionados (Elmasri e Navathe 2011).

Um banco de dados pode ser criado e mantido por um sistema gerenciador de banco de dados (SGBD), que acrescenta ao banco de dados e aplicações que fazem o uso dele, dá-se o nome de sistema de banco de dados (SBD) (Elmasri e Navathe 2011).

Sob a estrutura do banco de dados está o modelo de dados, um conjunto de ferramentas conceituais usadas para a descrição de dados, relacionamentos entre os dados, semântica de dados e regras de consistência. Dentre os modelos de dados propostos, pode-se destacar o modelo entidade-relacionamento e o modelo relacional, ainda muito utilizados (Silberschatz, Korth e Sudarshan 1999).

## **2.5. Computação Móvel**

Por volta de 1980, se tornou possível construir computadores pessoais leves o suficiente para serem carregados e que podiam ser conectados a outros computadores por meio de linhas telefônicas. Com a evolução tecnológica, a funcionalidade e o desempenho dos computadores tornou-se cada vez melhor, juntamente com a sua redução de tamanho.

Portanto, graças à miniaturização dos dispositivos e o avanço da conectividade sem fio, foi possível o surgimento da computação móvel. Dentre os dispositivos da computação móvel, tem-se:

- a) os aparelhos que cabem na mão, incluindo os telefones móveis “inteligentes” (*smartphones*);
- b) os assistentes digitais pessoais (*PDA's*);
- c) e outros equipamentos mais especializados operados manualmente (Coulouris 2013).

No entanto, a computação móvel apresenta diversos desafios para a sua construção, devido às plataformas móveis serem muito complexas, onde frequentemente seus usuários têm percepções muito diferentes dos recursos que esperam encontrar (Pressman 2016).

### **3. Trabalhos Relacionados**

AirBnb (2008) é um aplicativo desenvolvido para oferecer um serviço *online* comunitário para as pessoas anunciarem, descobrirem e reservarem acomodações e meios de hospedagem. O aplicativo apresenta a ideia de economia compartilhada, semelhante à proposta no trabalho, porém o mesmo é voltado para a área de hospedagem.

Cabify (2011) é um aplicativo desenvolvido pela empresa multinacional de rede de transporte Cabify, que oferece dois serviços para gerenciamento de transportes, um para empresas e outro para particulares. O Cabify se assemelha ao trabalho proposto, pois trata-se de serviço voltado à área de transporte e mobilidade urbana. Porém, o mesmo é mais voltado para prestação de serviços de táxi, incluindo taxas de cobrança e motoristas próprios, o que não é o foco do trabalho em questão.

Uber (2009) é um aplicativo desenvolvido pela empresa multinacional norte-americana Uber, que oferece um serviço semelhante ao táxi tradicional, conhecido popularmente como serviços de "carona remunerada". Assim como o Cabify, o Uber se assemelha ao trabalho proposto, pois trata-se de serviço voltado à área de transporte e mobilidade urbana, além de permitir o cadastro de motoristas autônomos. Porém, o mesmo também é mais voltado para prestação de serviços de táxi, incluindo taxas de cobrança, o que não é o foco do trabalho em questão.

Portanto, pode-se dizer que existem no mercado algumas aplicações que utilizam o conceito de economia compartilhada para as mais diversas áreas. No entanto, a aplicação construída nesse trabalho, se diferencia por focar mais na busca e divulgação do serviço de carona, e deixa os usuários livres para negociarem as condições do serviço de uma forma simples, prática e autônoma.

### **4. Metodologia**

Este trabalho apresenta o desenvolvimento de um aplicativo, onde alguns princípios da engenharia de *software* e metodologias ágeis de desenvolvimento são aplicados.

#### 4.1. Etapas do Trabalho

Este trabalho é dividido nas seguintes etapas:

- a) levantamento de requisitos;
- b) modelagem do sistema;
- c) desenvolvimento do sistema;
- d) teste do sistema.

No levantamento de requisitos foram definidas algumas funcionalidades pertencentes à aplicação.

A etapa de modelagem do sistema foi realizada com o uso de diagramas *UML*, para representar as interações entre os atores e as funcionalidades do sistema (Diagrama de Casos de Uso), assim como os relacionamentos entre classes (Diagramas de Classes). Nessa etapa também foi definida a arquitetura da aplicação, ou seja, com a mesma deve se comportar.

Na etapa de desenvolvimento, foram colocados em prática os conceitos definidos anteriormente nas etapas de especificação e projeto da aplicação, com o uso da metodologia ágil SCRUM como base.

Na fase de teste do sistema, houve a verificação se as funcionalidades definidas foram implementadas da forma especificada, assim como se a aplicação não apresenta inconsistências.

#### 4.2. Ambiente de Desenvolvimento

Neste trabalho, o ambiente de desenvolvimento é composto pelas seguintes ferramentas apresentadas na Tabela 1.

**Tabela 1. Ferramentas do ambiente de desenvolvimento.**

| Funcionalidade                         | Ferramenta                       |
|--|----------------------------------|
| Banco de Dados                         | MySQL                            |
| Diagramas UML                          | Astah Community                  |
| Programação Web (Servidor)             | Framework ASP .NET MVC 4         |
| Programação para Dispositivo (Cliente) | Framework Ionic v1 com AngularJS |
| Sistemas Operacionais                  | Windows 7 / 10                   |

## 5. Desenvolvimento

### 5.1. Levantamento de Requisitos

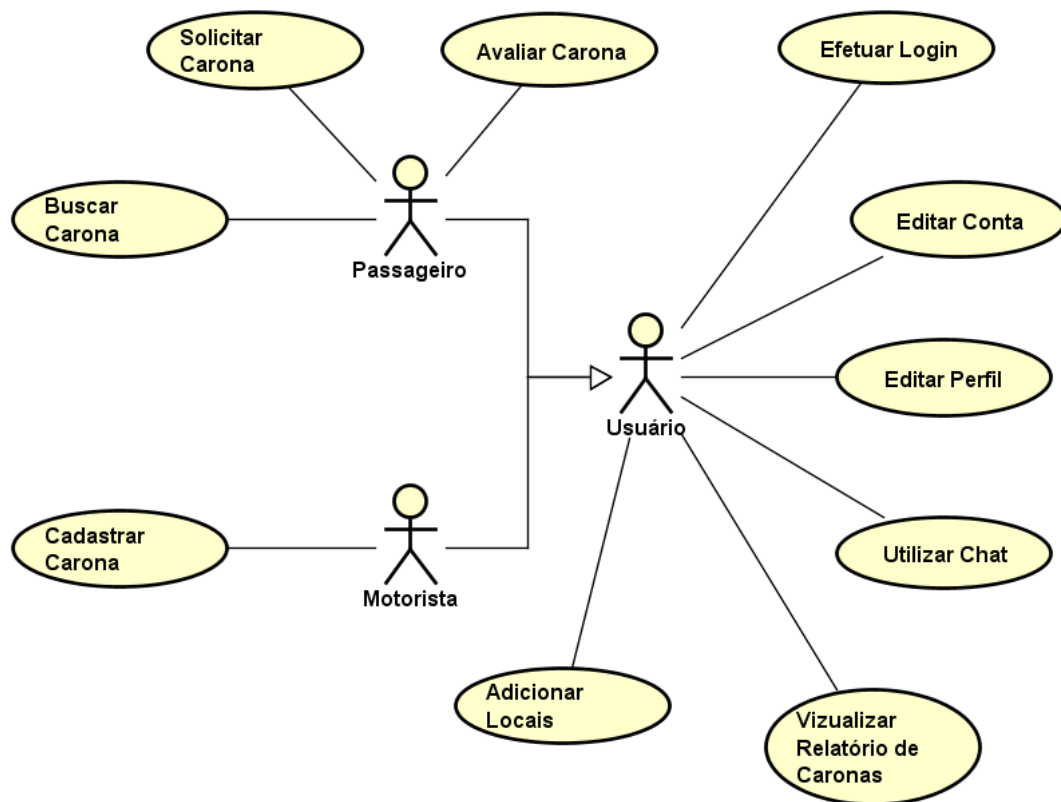
Com base nas necessidades observadas dos usuários, foram levantados os requisitos da aplicação e formalizados em um *Product Backlog* (um dos artefatos da metodologia ágil SCRUM), demonstrado na tabela a seguir:

**Tabela 2. *Product Backlog* da aplicação.**

| Product Backlog |                                 |  |            |             |
|-----------------|---------------------------------|--|------------|-------------|
| ID              | Título                          | Descrição  | Estimativa | Criticidade |
| 1               | Fazer Login                     | Permitir que o usuário faça login com a sua conta do Facebook ou conta de email.   | 20h        | Alta        |
| 2               | Editar Conta                    | Permitir que o usuário edite as informações da sua conta (nome, telefone, etc.).   | 10h        | Alta        |
| 3               | Editar Perfil                   | Permitir que o usuário edite as informações do seu perfil (motorista ou passageiro).   | 10h        | Alta        |
| 4               | Adicionar Locais                | Permitir que o usuário cadastre locais em que vá com mais frequência (como casa, trabalho).  | 10h        | Baixa       |
| 5               | Cadastrar Carona                | Permitir que o usuário (motorista) insira seu destino, saída e hora de saída, além do cadastro de vagas e modelo do carro.                                     | 30h        | Alta        |
| 6               | Buscar Carona                   | Permitir que o usuário (passageiro) receba uma lista de usuários (motoristas) que possuem destino comum (de acordo com os filtros), e que tenha vaga no carro. | 30h        | Alta        |
| 7               | Solicitar Carona                | Permitir que o usuário (passageiro) escolha e solicite a carona de preferência, da lista de caronas recebidas.   | 30h        | Alta        |
| 8               | Gerar Notificação ao Motorista  | Permitir que o usuário (motorista) receba uma notificação quando algum usuário (passageiro) escolhe a sua carona.  | 10h        | Alta        |
| 9               | Utilizar Chat                   | Permitir que os usuários (motorista e passageiro) se comuniquem via chat para entrarem em mais detalhes sobre a carona.  | 20h        | Média       |
| 10              | Avaliar Carona                  | Permitir que o usuário (passageiro) possa avaliar a carona escolhida.  | 5h         | Média       |
| 11              | Vizualizar Relatório de Caronas | Permitir que os usuários possam vizualizar as caronas abertas e recebidas por ele.   | 10h        | Média       |

### 5.2. Modelagem do Sistema

As figuras abaixo apresentam a modelagem *UML* e a arquitetura da aplicação. A Figura 1 apresenta uma visão geral do Diagrama de Casos de Uso, ou seja, considera as funcionalidades básicas em que os atores interagem com a aplicação, com base no *Product Backlog* levantado anteriormente:



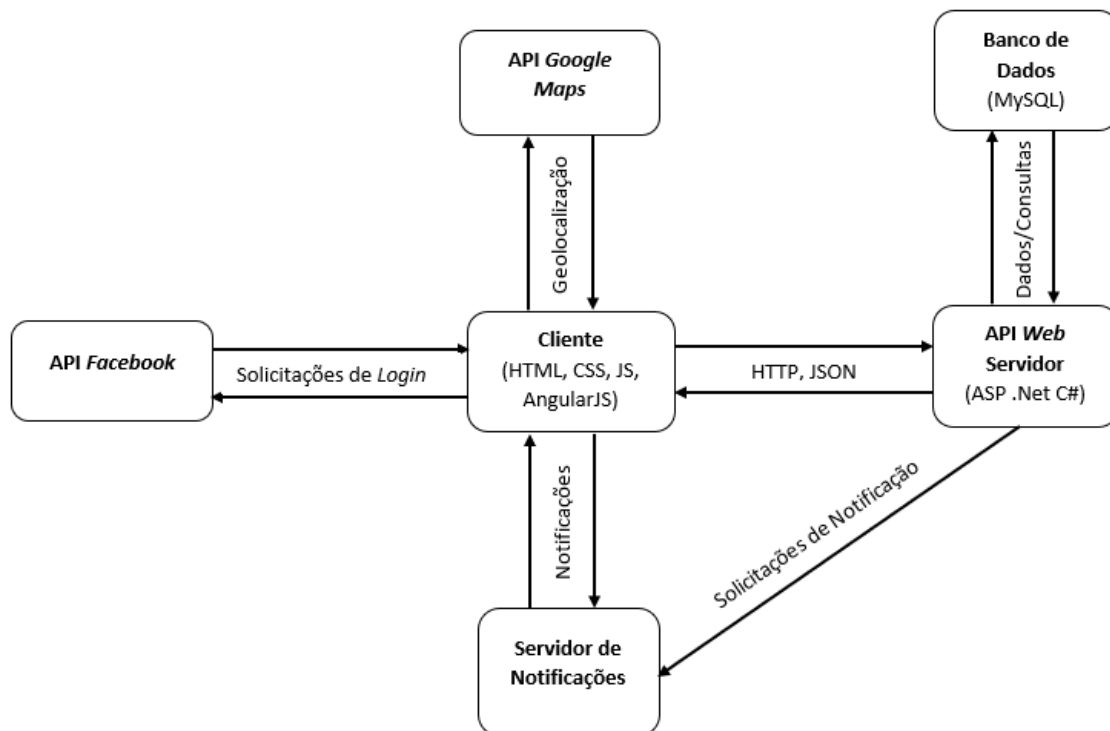
**Figura 1. Diagrama de Casos de Uso geral**

A Figura 2 apresenta o Diagrama de Classes que descreve a estrutura da aplicação, onde são modeladas as suas classes, atributos, operações e relacionamentos entre os objetos. A partir dele, fica mais fácil definir como o sistema deve funcionar internamente, assim como, facilita também a modelagem do banco de dados da aplicação:





- uso da aplicação é registrado nesse serviço, e esse registro é armazenado pelo servidor da aplicação para o mesmo requisitar as notificações;
- b) um serviço de autenticação utilizando os dados do *Facebook* do usuário armazenados no dispositivo;
  - c) um serviço de geolocalização da Google, para encontrar localizações, cálculo de rotas, visualização de mapas, etc.



**Figura 3. Arquitetura da aplicação**

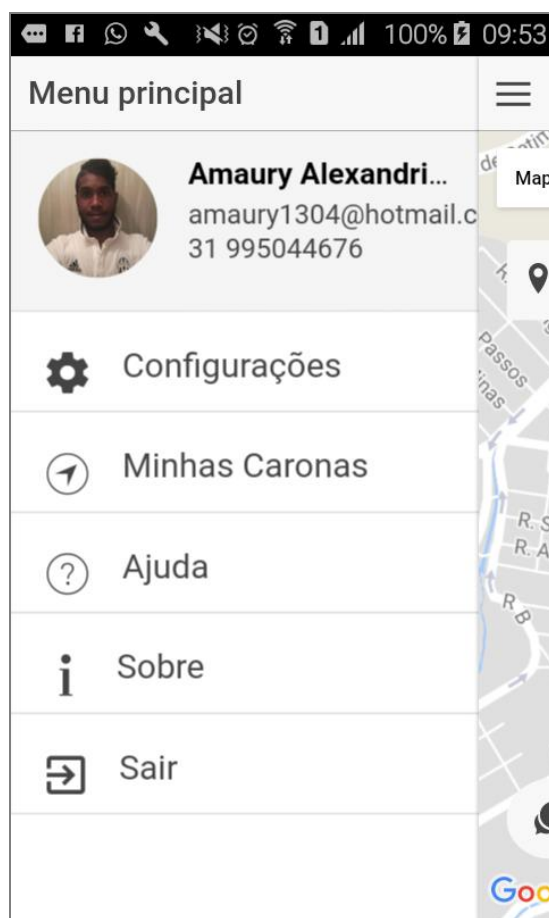
### 5.3. Interface do Sistema

As figuras a seguir apresentam as principais telas de interface da aplicação. A Figura 4a mostra a tela principal que aparece logo após que o usuário é autenticado na aplicação. A partir dela, pode-se navegar no sistema, de forma que são dispostas algumas funcionalidades em forma de opções apresentadas em um menu lateral, conforme mostrado na Figura 4b:





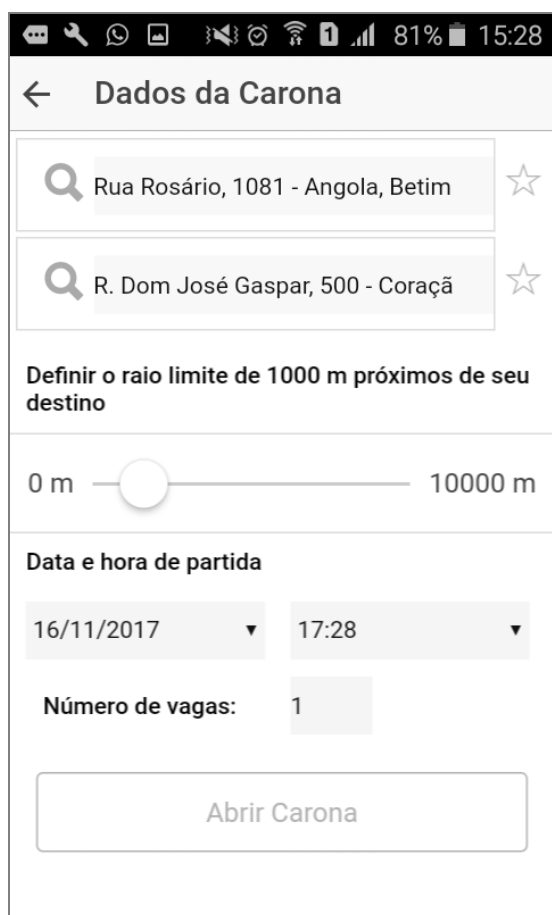
**Figura 4a. Tela principal da aplicação**



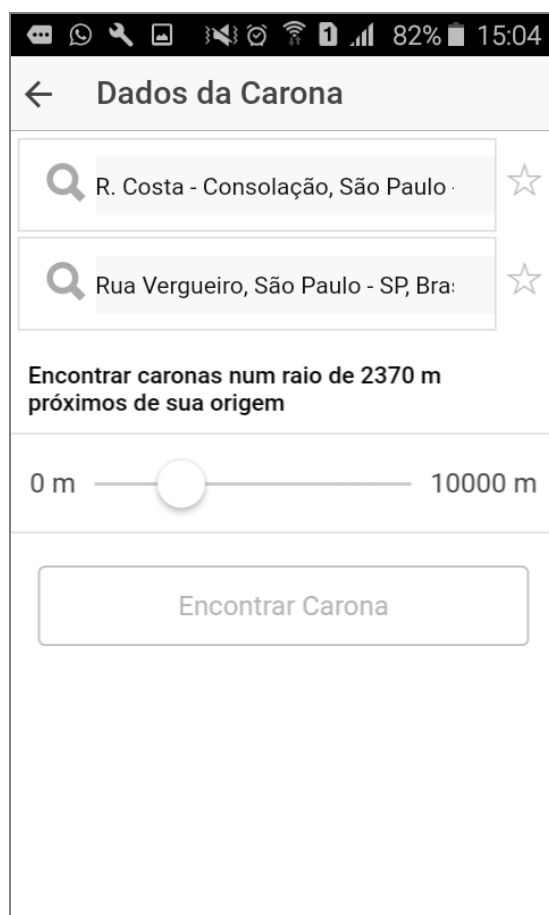
**Figura 4b. Menu lateral**

Ao decidir abrir uma carona, será apresentada uma tela ao usuário motorista para preenchimento dos dados da carona, conforme a Figura 5a. Da mesma forma, caso o usuário seja passageiro e esteja à procura de uma carona, será apresentada também uma tela para preenchimento dos dados solicitados, conforme a Figura 5b:





**Figura 5a. Tela para abertura de carona**



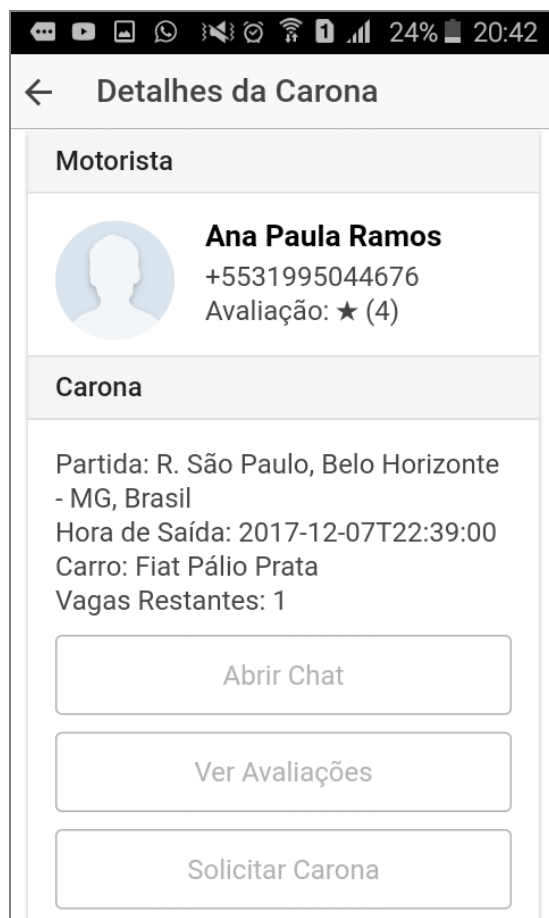
**Figura 5b. Tela para solicitação de carona**

Ao procurar uma carona, são retornados ao passageiro as opções de caronas encontradas (em forma de lista e em forma de ícones no mapa), de acordo com filtros definidos anteriormente, conforme a tela mostrada na Figura 6a. Também, o usuário tem a opção de ver mais detalhes sobre a carona ao clicar na mesma, conforme a tela mostrada na Figura 6b.





**Figura 6a. Mapa com as caronas encontradas**

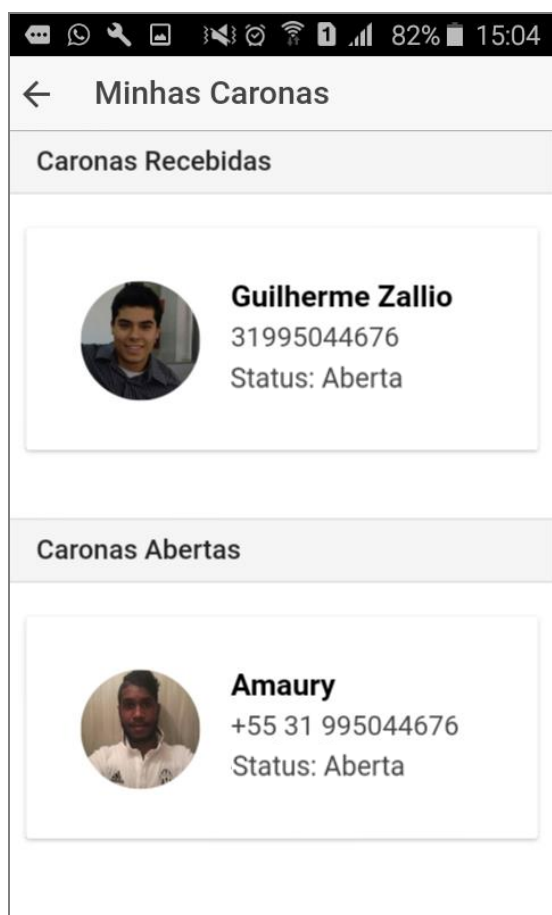


**Figura 6b. Tela para visualização dos detalhes da carona**

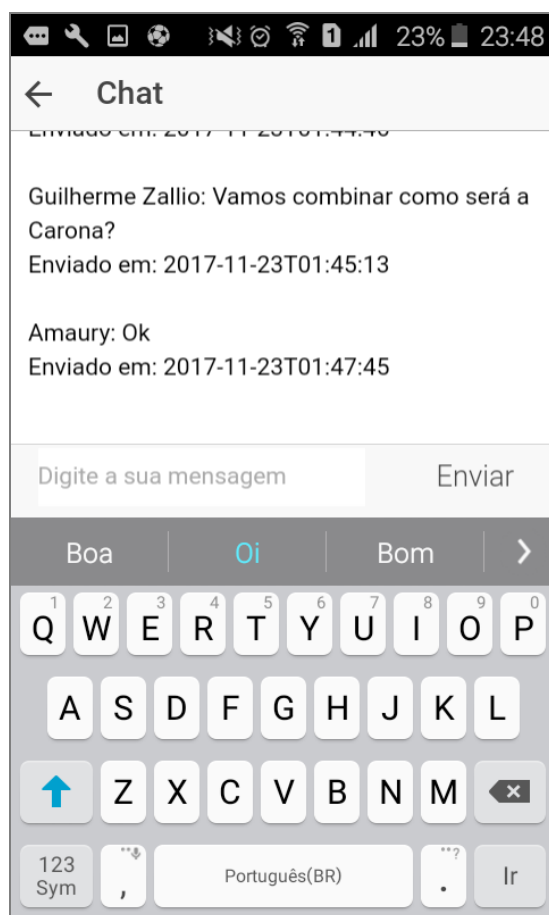
Além disso, a aplicação conta com outras funcionalidades adicionais, e permite por exemplo, que seja exibido um histórico de caronas do usuário, como mostrado na Figura 7a, assim como, possa ser utilizada uma ferramenta de *chat* para comunicação direta entre quem está oferecendo e quem está solicitando a carona, como mostrado na Figura 7b:







**Figura 7a. Histórico de caronas do usuário**



**Figura 7b. Tela de chat**

## 7. Conclusão

Este trabalho apresentou o desenvolvimento de um aplicativo que utiliza uma plataforma híbrida de desenvolvimento para dispositivos móveis, além de *API's* (*Application Programming Interface*) baseadas em nuvem para guardar e disponibilizar os dados. Os resultados mostraram que a maioria das funcionalidades do sistema proposto foram desenvolvidas, deixando algumas delas para um desenvolvimento futuro, como aprimoramento do sistema. Porém, as que foram desenvolvidas são úteis, funcionam corretamente e são parte importante para o funcionamento do mesmo.

O aplicativo não foi disponibilizado para ser utilizado pelos usuários. Apenas foram feitos testes unitários a fim de testar as funcionalidades e certificar que em suas essências, elas funcionam corretamente. Ele não foi testado em larga escala, ou seja, não foi feito um teste de carga, então não se sabe se o aplicativo reagiria bem a essa situação específica. No entanto, o aplicativo está preparado para entrar em funcionamento, precisando apenas de ajustes na estrutura que hospeda os seus serviços, para que consiga ser utilizado em larga escala.

A principal contribuição futura desse trabalho é que, quando o aplicativo for implantado, irá auxiliar a tornar mais acessível um tipo de transporte alternativo, que é a carona. Também permitirá a aproximação das pessoas, ao contribuir para a socialização das mesmas.

Como trabalhos futuros, pretende-se melhorar o mecanismo de bate-papo, que hoje consome um número muito alto de requisições (o que atrapalha o desempenho da aplicação) para ficar mais usual e confortável ao usuário. Também pretende-se implementar uma funcionalidade para que o motorista avalie os seus passageiros, pois como se trata de uma situação de confiança de ambas as partes, se faz necessário um leque maior de ferramentas para armazenar as informações dos usuários. Por fim, pretende-se acrescentar a possibilidade de armazenar a cópia das fotos e imagens advindas dos bate-papos e dos cadastros de usuários e carros em um servidor próprio para poder exibi-las no aplicativo. Também pretende-se armazenar áudios e vídeos adicionados no bate-papo.

## **Referências Bibliográficas**

- Airbnb, (2008), “Airbnb”, <https://www.airbnb.com.br/>, Acesso em 10 set.
- Audy, Jorge Luis Nicolas, (2011) “Fundamentos de sistemas de informação”. Porto Alegre, Bookman.
- Bezerra, Eduardo, (2003) “Princípios de análise e projeto de sistemas com UML”, Rio de Janeiro, Campus.
- Cabify, (2011), “Cabify para Motoristas”, <https://cabify.com/pt-BR/drivers>, Acesso em 10 set.
- Coulouris, George F. et al, (2013) “Sistemas distribuídos: conceitos e projeto”, Porto Alegre, Bookman.
- Elmasri, Ramez; Navathe, Shamkant B., (2011) “Sistema de Banco de Dados”, São Paulo, Pearson.
- Fowler, Martin, (2011) “UML essencial um breve guia para linguagem padrão”, Porto Alegre, Bookman.
- Laudon, Kenneth C.; Laudon, Jane Price, (2011) “Sistemas de informação gerenciais”, São Paulo, Pearson.
- PMI, (2013), “UM GUIA do conhecimento em gerenciamento de projetos: guia PMBOK”, São Paulo, Saraiva.
- Pressman, Roger, (2016) “Engenharia de software”, Porto Alegre, AMGH.
- Sebrae, (2017) “Economia Colaborativa: a tendência que está mudando o mercado”, <http://www.sebrae.com.br/sites/PortalSebrae/artigos/economia-colaborativa-a-tendencia-que-esta-mudando-o-mercado,49115f4cc443b510VgnVCM1000004c00210aRCRD>, Acesso em 10 jun.
- Silberschatz, Abraham, Korth, Henry F. e Sudarshan, S., (1999) “Sistema de banco de dados”, São Paulo, Pearson Education do Brasil.
- Sommerville, Ian, (2011) “Engenharia de Software”, São Paulo, Pearson 548.
- Uber, (2009), “Uber: Inscreva-se para dirigir ou toque para viajar”, <https://www.uber.com/pt-BR/>, Acesso em 10 set.