

FOLLOW US: APLICATIVO PARA COMPARTILHAMENTO DE LOCALIZAÇÃO EM TEMPO REAL

Lucas Scherer Garcia¹

Prof. Giovani Nardi²

Resumo: Saber a localização de determinada pessoa ou grupo de pessoas em tempo real sempre foi algo desejável, tanto no mundo empresarial quanto na vida particular das pessoas. Com a popularização de smartphones, dispositivos móveis que possuem diversos recursos dentre eles acesso a internet e GPS (*Global Positioning System*), isto se tornou possível. Muitos usuários utilizam aplicações georreferenciadas para compartilhar sua localização com seus contatos. No entanto, muitas dessas aplicações utilizam protocolos de comunicação que precisam constantemente acessar o servidor para sincronizar o aplicativo com novas informações, consumindo desnecessariamente o pacote de dados do usuário em redes móveis. O Follow Us, aplicativo que é resultado deste trabalho de pesquisa, possibilita como recurso principal o compartilhamento de localização entre grupos de contatos em tempo real com alta precisão e frequência nas atualizações, utilizando um framework apropriado para realizar a comunicação com o servidor.

Palavras-chave: Localização. Grupos. Tempo real. Internet. Smartphones.

Abstract: Knowing the location of a given person or a group in real-time has always been a subject of desire, both in business and in people's private life. With the popularization of smartphones and mobile devices that have several resources with internet access and GPS (*Global Positioning System*) among them, it became possible. Many users make use of georeferenced applications to share their location with their contacts. However, many of these applications use communication protocols that need to constantly access the server to synchronize the application with new data, unnecessarily consuming the user's mobile data package in mobile

¹ Graduado em Web Design e Computação, Unisul, Pós-Graduando em Desenvolvimento de Aplicações para Dispositivos Móveis, Unisinos. Email: garcia.lucass@terra.com.br

² Orientador. Email: giovani@gscontrol.com.br

networks. The Follow Us, application that is the result of this research enables, as the main feature, sharing the location in groups in real-time with high precision and frequency in the updates, using an appropriate framework to make the communication with the server.

Key-words: Location. Groups. Real-time. Internet. Smartphones.

1 INTRODUÇÃO

Não é novidade que a popularidade dos *smartphones* cresceu rapidamente nos últimos anos. Com este crescimento muitos recursos como acesso à internet, recursos de multimídia, armazenamento de dados, criação de conteúdo e navegação pelo GPS (em português, Sistema de Posicionamento Global) e outras tecnologias que estavam disponíveis apenas separadamente em outros tipos de dispositivos agora estão ao alcance de toda a população por preços acessíveis.

Surgiram então, diversas oportunidades de explorar estes recursos de forma a integrá-los harmoniosamente para solução de problemas de um usuário comum, antes inviáveis senão do ponto de vista técnico, então na questão financeira, visto que muitas *features* que agora são possíveis com *smartphones*, eram somente encontradas em dispositivos desenvolvidos para fins militares, como é o caso de aparelhos que permitem o compartilhamento de localização em tempo real.

Apesar de oferecer diversos recursos, os *smartphones* ainda são um tanto limitados no que diz respeito ao seu poder computacional e capacidade de armazenamento além de o tráfego de dados pelas redes móveis nem sempre ser estável e/ou confiável. Isto pode oferecer dificuldades técnicas ao desenvolvedor de aplicações para este tipo de dispositivo, pois o mesmo deve se preocupar em como oferecer uma experiência suave ao usuário final e lidar, em tempo de projeto, com estas adversidades, como por exemplo, a já citada instabilidade e limitação no envio e recebimento de informações pelo dispositivo.

Com base nas informações apresentadas acima, este trabalho pretende apresentar uma aplicação para a plataforma *Android* que possibilite que os usuários

possam compartilhar sua localização em tempo real com grupos definidos por ele. Muitas vezes, tal funcionalidade é extremamente útil. Podemos citar diversos casos de uso em que esta utilidade é provada. Por exemplo, um professor que está em excursão com alunos e gostaria de manter-se informado da localização de todas as pessoas sob sua tutela. Ou ainda pais que desejem saber a velocidade a que chega o transporte escolar de seus filhos em tempo real, sendo também notificado em quando perímetros pré-estabelecidos foram deixados ou adentrados.

Ainda levando em consideração as limitações dos *smartphones*, acima descritas, é crucial que a aplicação desenvolvida utilize um *framework* que possibilite que um servidor de aplicação não somente receba informações, mas que também consiga enviá-las para os usuários conectados a ele, proporcionando uma experiência suave ao usuário final, pois evita acessos desnecessários a rede de dados móveis.

2 REFERENCIAL TEÓRICO

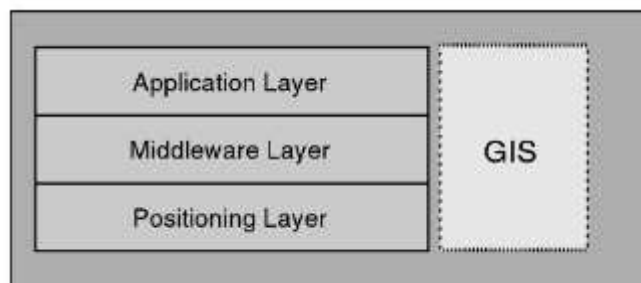
Esta seção apresenta os conceitos utilizados no desenvolvimento deste artigo.

2.1 Location Based Services

Basicamente *Location Based Services* (Serviços Baseados em Localização) é um conceito utilizado para definir as aplicações que utilizam coordenadas de localização geográfica para reagir de acordo com a mesma. Podemos citar como exemplo aplicações que utilizam a posição atual do usuário para exibir informações relevantes em sua proximidade, como hospitais, parques, postos de polícia, restaurantes, bares ou qualquer outro tipo de estabelecimento. De acordo com Sarah Spierkermann em *Location-Based Services* (2004), serviços de localização podem ser definidos como serviços que integram a localização ou posição com outras informações para proporcionar mais valor para o usuário. Tradicionalmente, a informação destes serviços é obtida através de um dispositivo com a ajuda de um sistema de satélite GPS, porém existem outras formas, como triangulação com a rede móvel. São usados em três principais áreas: indústrias militares e governamentais, serviços de emergência e no setor comercial.

Embora os *Location Based Services* (LBS) sejam geralmente relacionados a dispositivos móveis, eles não são necessariamente utilizados neste tipo de dispositivo. *Location Based Services* são utilizados para melhorar os algoritmos de busca na web, para navegação em informações de trânsito, bens e serviços como também para localizar outros usuários deste tipo de serviço (WILSON, 2012), portanto um LBS pode ser também considerado um tipo peculiar de Web Service, conforme observado por Schiller (2004).

Figura 1- Arquitetura básica de um LBS



Fonte: Schiller (2004)

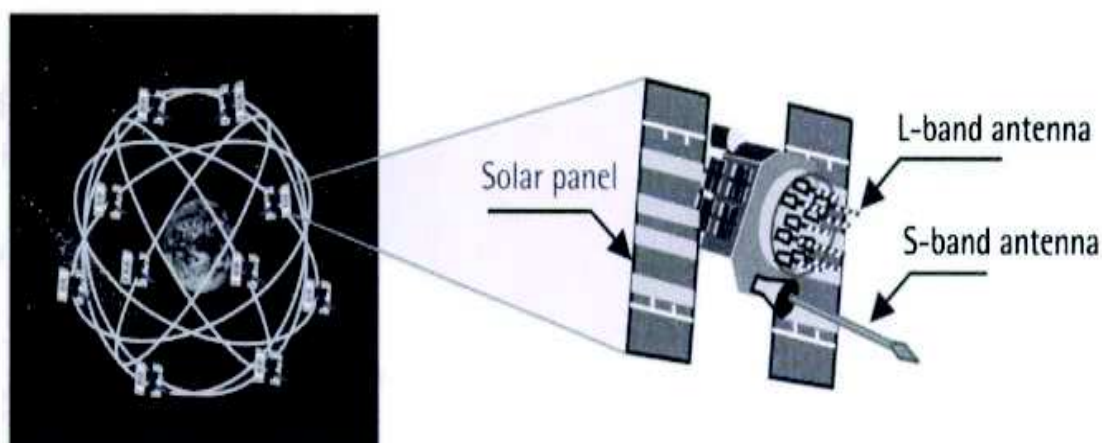
2.1.1 GPS

De acordo com El-Rabbany (2002), o sistema global de posicionamento (GPS) é um sistema de navegação baseado em satélite que foi desenvolvido pelo Departamento de Defesa dos Estados Unidos no início da década de 1970. Inicialmente, o GPS foi desenvolvido como um sistema militar para atender as forças armadas dos Estados Unidos. No entanto, mais tarde foi disponibilizado para os civis, e é agora um sistema de uso duplo, que pode ser acessado tanto por militares e usuários civis. Conforme explicado por El-Rabbany (2002), o sistema de GPS fornece informações contínuas de posição e horário em qualquer lugar do mundo sob qual condição meteorológica. Atualmente a constelação do GPS possui pelo menos 24 satélites totalmente operacionais.

O uso deste sistema abrange um grande leque de aplicações. Seu uso pode ser terrestre, aéreo ou marítimo. É muito utilizado em *Location Based Services*, visto que pode oferecer informações de localização de forma muito precisa, com margem de

erro de apenas três metros. Grande parte dos smartphones da atualidade já possui um receptor do sinal dos satélites de GPS.

Figura 2- Constelação de GPS



Fonte: El-Rabbany (2002)

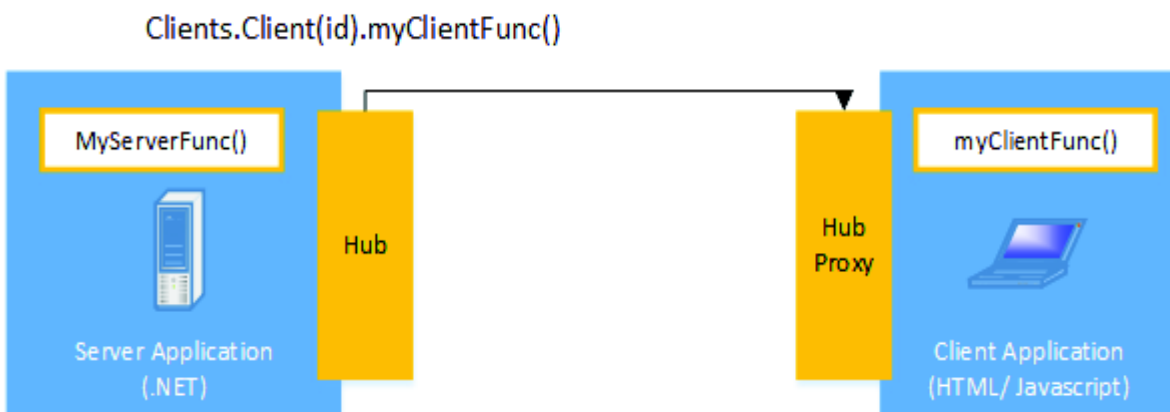
2.2 SignalR

De acordo com Fletcher (2014), *SignalR* é uma biblioteca que simplifica o processo de implementação de funcionalidades em tempo real em aplicações. Funcionalidades em tempo real é a habilidade de ter código do lado do servidor enviando conteúdo para clientes instantaneamente conforme fica disponível, ao invés de o servidor aguardar o cliente solicitar novas informações. Ainda de acordo com Fletcher (2014), a biblioteca *SignalR* lida com o gerenciamento de conexões automaticamente, e permite que as mensagens sejam transmitidas simultaneamente para todos clientes conectados, como por exemplo, um chat. É um framework *open-source* da Microsoft e é baseado na plataforma .NET, possuindo também uma implementação para *Xamarin*, uma extensão para a IDE Visual Studio, que permite o desenvolvimento de aplicações para *Android*, *iOS* e *Windows*.

Esta biblioteca permite ao desenvolvedor utilizar três tipos principais de transportes de conexão. Porém é possível deixar a configuração automática, ou

seja, ele avalia qual é a melhor opção para o cliente ao realizar a conexão. Se tanto o cliente (como é o caso das principais plataforma móveis hoje como *Android*, iOS e Windows) quanto o servidor suportarem o uso de *WebSockets*, este é o método selecionado, por ser o mais eficiente para atualizações de alta frequência. No entanto, se não for possível utilizar *WebSockets*, a biblioteca automaticamente escolhe outro transporte como *Server Sent Events* ou *Long-Polling*.

Figura 3- Aplicação cliente invocando método do servidor



Fonte: Fletcher (2014)

2.3 WebSockets

Websockets é um protocolo de comunicação que permite que seja estabelecida uma conexão bidirecional entre cliente e servidor. Com isto, é possível que seja aberta uma conexão TCP/IP sobre o browser, possibilitando que o servidor também consiga enviar informações aos clientes conectados, diferente da arquitetura tradicional onde apenas o cliente faz a requisição e o servidor responde com conteúdo. Apesar de ser uma tecnologia originalmente desenvolvida para browsers, pode ser utilizada por qualquer aplicação cliente que implemente o protocolo.

Uma conexão *WebSockets* inicia com uma requisição HTTP, cujo cabeçalho possui um pedido de *update* do protocolo e uma resposta do servidor indicando se foi possível realizar esta troca. Esta operação é denominada *handshake*. Após o *handshake*, a comunicação entre o cliente e o servidor passa a ser feita sobre um

único soquete, não sendo mais necessárias requisições HTTP. É importante ressaltar que, pelo fato de funcionar através de uma conexão TCP/IP, pode haver problemas com bloqueio em firewalls.

Figura 4- Resposta do servidor após solicitação de upgrade do protocolo

```
▼ General
Request URL: ws://echo.websocket.org/echo
Request Method: GET
Status Code: 200 OK [text/plain]
Response Headers
view source
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: x-websocket-protocol
Access-Control-Allow-Headers: x-websocket-version
Access-Control-Allow-Headers: x-websocket-extensions
Access-Control-Allow-Headers: authorization
Access-Control-Allow-Headers: content-type
Access-Control-Allow-Origin: null
Connection: Upgrade
Date: Wed, 22 Jun 2016 18:29:13 GMT
Sec-WebSocket-Accept: Cxf7q5GfveCFQ9VEmtr00vKvNP0=
Server: Kaazing Gateway
Upgrade: websocket
```

Fonte: Elaborado pelo autor

2.4 Long-polling

Long-polling é o termo utilizado para definir o processo de obtenção de informações do servidor usando o modelo tradicional de requisições HTTP. Nesta arquitetura, a aplicação cliente envia uma requisição HTTP e o servidor mantém a mesma em aberto (com um longo *timeout*, tempo limite para a conexão ser fechada) até o momento em que o servidor possui novas informações, quando os dados são disponibilizados e a conexão fechada. Após o encerramento da conexão, uma nova conexão é novamente aberta, iniciando novamente o processo.

2.5 Server Sent Events

Server Sent Events (SSE) é uma tecnologia que permite que o servidor envie informações para os clientes conectados através de uma conexão sobre o protocolo HTTP. Diferentemente do *Long-Polling*, a conexão não é finalizada após o cliente receber as informações. Ao invés disso, a conexão fica aberta, possibilitando que dados continuem trafegando entre cliente e servidor.

2.6 Geocerca

Geocerca é um acrônimo de duas palavras: "cerca", que significa um recinto ou perímetro, e o prefixo "Geo" que significa que este perímetro é construído com dados geográficos. Geocerca pode ser definida como a delimitação de uma área geográfica usando um perímetro virtual (MATERA; ROSSI, 2013). De acordo com Chen e Guinness (2014), geocerca é o processo de monitorar se um usuário ou dispositivo atravessou um região especificada. A região pode ser definida por uma forma geométrica ou um círculo, ou em alguns casos por um segmento de linha.

Conforme explicado por Chen e Guinness (2014), as geocercas são mais comumente utilizadas para rastreamento de ativos e gerenciamento de frotas como, por exemplo, uma aplicação que registra sempre que um caminhão entra ou sai de uma área de carregamento. Também podem ser utilizadas para aplicações de proteção e segurança para alertar, por exemplo, se uma pessoa monitorada como algum paciente que sofre de demência abandona uma área controlada.

3 TRABALHOS RELACIONADOS

O compartilhamento de localização em tempo real é algo bastante útil para a sociedade, visto que este recurso é utilizado nas mais diversas áreas de conhecimento ou aplicação. Podemos citar diversos exemplos como empresas que utilizam rastreamento em seus veículos ou Estado quando utiliza este recurso para monitorar forças de segurança. Na área médica o uso de localização em tempo real pode ser muito útil para monitorar pessoas que sofrem de Alzheimer, geralmente idosos. Também é importante citar casos onde os pais querem, por questões de

segurança, ter controle de seus filhos menores através deste tipo de monitoramento da localização e outras informações como a velocidade do transporte escolar.

Existem aplicativos para dispositivos móveis no mercado que conseguem prover este tipo de monitoramento. Um deles chama-se *Life360*. No *Life360*, é possível criar círculos nos quais é possível adicionar pessoas. Através destes círculos, é possível compartilhar a localização com todas as pessoas inclusas no mesmo. Também é possível adicionar locais, que são utilizados para alertar quando alguém entra ou sai de determinada região. O aplicativo *Follow Us* que é fruto desta pesquisa, além de oferecer estas funcionalidades como criação de círculos (denominado como "grupo" no *Follow Us*) e alertas de entrada ou saída de determinada região, também possibilita que a velocidade do usuário, em quilômetros por hora, seja visualizada em tempo real. Também é importante citar que o *Life360* utiliza o serviço de mensagens do Google, o *Google Cloud Messaging* para enviar os dados através da rede. Neste aspecto, do ponto de vista técnico, o *Follow Us* se sobressai, visto que utiliza uma infraestrutura própria de comunicação, não ficando dependente de serviços de terceiros para operar, o que pode ser crítico em determinadas situações.

Outro aplicativo que também pode ser comparado com o *Follow Us* é um aplicativo chamado *Glympse*. Este aplicativo permite o compartilhamento de localização em tempo real, porém em um contexto mais individual. Basicamente, o usuário selecionará a opção de compartilhar localização e enviará um link para os contatos selecionados. Ao receber o link, os destinatários poderão visualizar em um mapa o deslocamento do remetente do link. O aplicativo fruto desta pesquisa se diferencia no que diz respeito aos grupos de localização. Com o *Follow Us*, é possível ver a localização de todas as pessoas de um grupo ao mesmo tempo além de ser possível a criação de locais para que os demais participantes saibam quando alguém entrou ou saiu de determinada região, como já explicado acima.

4 METODOLOGIA DE PESQUISA

De acordo com Lima (2004), a pesquisa científica é caracterizada da seguinte maneira:

A pesquisa científica, qualitativa ou quantitativa, pressupõe, como atividades fundamentais: a identificação do objeto de pesquisa, a coleta de dados, o tratamento e análise dos dados e a comunicação dos resultados à comunidade. O que difere uma pesquisa de natureza quantitativa de uma qualitativa são os recursos técnicos envolvidos e os procedimentos metodológicos, sobretudo as técnicas de coleta de dados e os métodos de tratamento e análise dos materiais reunidos.

Conforme explicado, nesta pesquisa, o problema a ser resolvido consiste basicamente no desenvolvimento de uma aplicação para dispositivos móveis que permita o compartilhamento da localização geográfica em tempo real para um ou diversos grupos de pessoas. Além disso, também é objetivo deste trabalho, elaborar uma aplicação que reúna funcionalidades que existem de forma separada em outras soluções existentes no mercado, conforme explicado no capítulo anterior. Esta problemática foi observada a partir da análise de uma situação real onde o compartilhamento de localização em tempo real se fez necessário, porém não haviam sido encontradas soluções ideais para a situação. Também foi levado em conta o fato de que as soluções existentes para smartphones não agregavam todas as funcionalidades que se faziam necessárias para a problemática identificada.

A pesquisa iniciou-se com o estudo de assuntos relacionados a esta problemática, passando por temas como o funcionamento do GPS e protocolos e *frameworks* de comunicação. Tal estudo se fez necessário para que as dificuldades técnicas pudessem ser analisadas e proposta a solução, ou seja, um aplicativo para *Android*. Depois de criado o aplicativo, o mesmo foi distribuído para um grupo de pessoas para que estas fizessem uma avaliação do funcionamento do mesmo.

No caso tipo de abordagem de pesquisa, foi escolhida a abordagem quantitativa, pois de acordo com Gressler (2004), esta abordagem tem, em princípio, a intenção de garantir a precisão dos resultados, evitar distorções de análise e interpretação. Desta forma, é possível verificar o grau de sucesso da solução encontrada para a problemática. Para a obtenção destes resultados, foi necessário lançar mão de métodos de pesquisa e coleta de dados. O método utilizado foi o *survey*, pelos motivos explicados por Lima (2004):

(...) é atualmente, o que melhor representa as características da pesquisa quantitativa, e isto porque corresponde a uma abordagem de fenômeno investigado envolvendo a realização de uma pesquisa de

campo, na qual a coleta de dados é feita por meio de aplicação e questionário e/ou formulário junto à população alvo da pesquisa.

O questionário foi elaborado de forma a obter a opinião do público alvo de forma mais fiel possível à realidade. As perguntas deste questionário tinham como objetivo avaliar a importância das funcionalidades do aplicativo, o desempenho e a facilidade encontrada na utilização do mesmo do ponto de vista do usuário final. Além disso, as respostas foram dadas em graus numéricos de acordo com a escala de Likert, desenvolvida pelo psicólogo Rensis Likert, para mensurar fielmente as opiniões das pessoas em relação ao tema em questão. O questionário foi aplicado no Grupo de Escolta Olímpica da Polícia Rodoviária Federal, constituído de policiais motociclistas, após os mesmos terem participado dos testes do aplicativo, conforme descrito no capítulo 6.

5 DESENVOLVIMENTO DO APLICATIVO

Nesta seção será descrito o processo de desenvolvimento do Follow Us, aplicativo desenvolvido para materializar esta pesquisa.

5.1 Processo de desenvolvimento

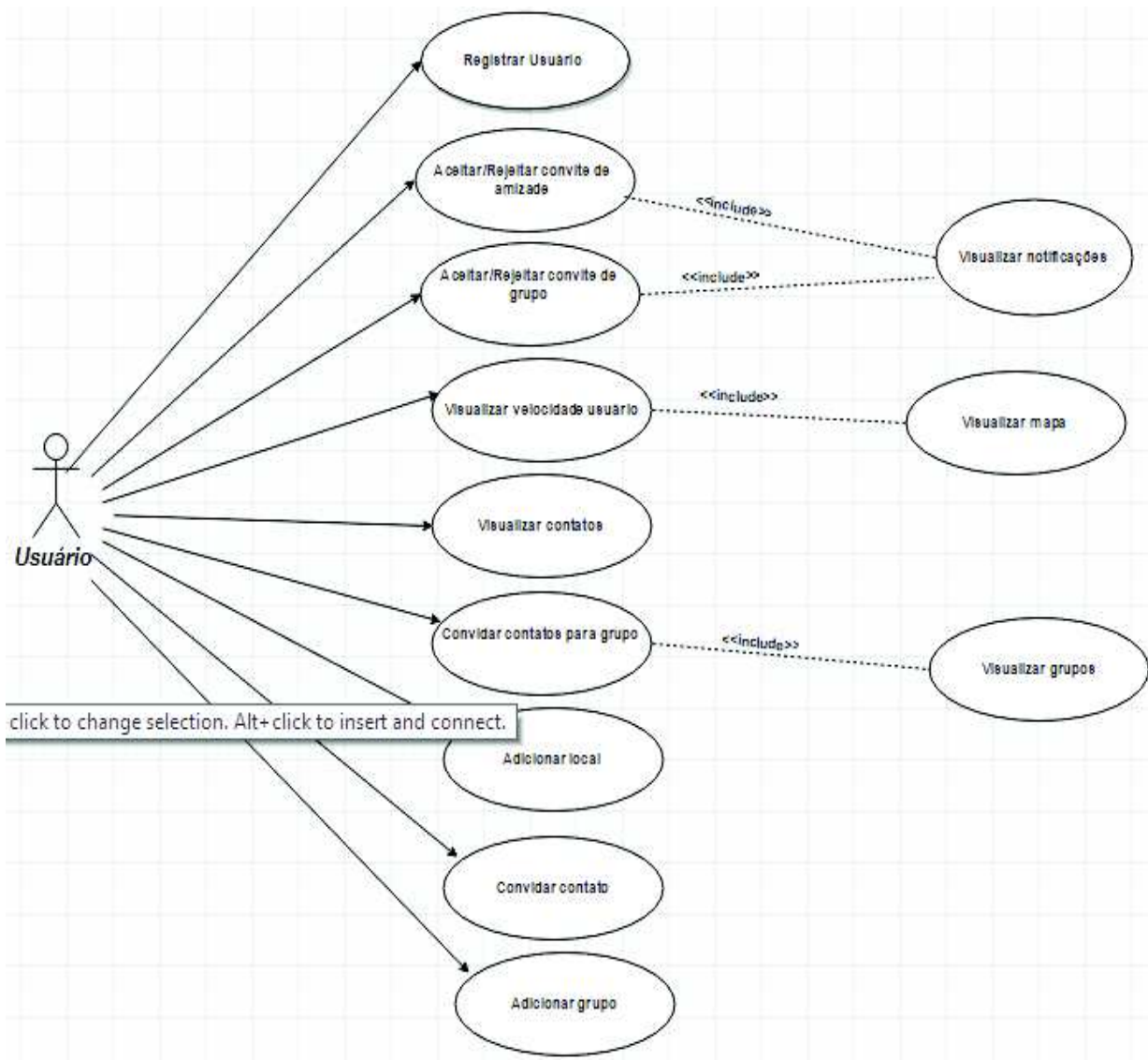
O primeiro passo para o desenvolvimento do aplicativo foi a definição dos requisitos funcionais e não funcionais da aplicação, ou seja, as funcionalidades que a mesma deveria contemplar. Depois de feitas estas definições, foi definida a arquitetura a ser utilizada e as tecnologias, como a IDE e linguagem de programação, sistema gerenciador de banco dados, frameworks de persistência de dados e métodos de autenticação.

5.2 Requisitos

De acordo com Martins (2007) os requisitos são características funcionais e não funcionais que o sistema precisa apresentar. Os requisitos funcionais são aqueles que definem o comportamento do sistema, capturado através de casos de uso, que documentam as entradas, os processos e as saídas geradas. Para apresentar os requisitos foi criado um diagrama de casos de uso, conforme figura 5,

já que este diagrama é de simples entendimento inclusive para leigos na área, visto que mostra um panorama geral do sistema.

Figura 5 - Diagrama de casos de uso



Fonte: Elaborado pelo autor

Já os requisitos não-funcionais são as características de um sistema no que diz respeito a usabilidade, performance e confiabilidade. Os requisitos não-funcionais do Follow Us são:

- a) Salvar os dados da conta do usuário, como os contatos e grupos em um banco de dados local, e sincronizar os dados quando o usuário se conecta novamente. Desta forma, o usuário pode visualizar as informações em cache, quando estiver desconectado.
- b) Após feito o login pela primeira vez, o aplicativo salva as credenciais para posterior acesso..
- c) O aplicativo deve iniciar após o boot do sistema operacional *Android*.
- d) O aplicativo deve possuir versão traduzida para ingles, além do português.

5.3 Funcionamento da aplicação

Primeiramente a aplicação deve ser instalada no dispositivo *Android* através da loja de aplicativos da Google, a *Play Store*. Para isto, foi criado um aplicativo na conta do *Google Developer Console* do autor deste artigo com o nome de "Follow Us", onde foi feito o upload do pacote de instalação do mesmo. Para manter um controle sobre quem estava utilizando o aplicativo, a aplicação foi configurada inicialmente para Alfa fechado, onde somente as pessoas com e-mail cadastrado em uma lista específica e que possuíam o link de instalação poderiam fazer download. O motivo desta configuração inicial é que funcionalidades e otimizações de desempenho em relação ao servidor, como o acesso a banco de dados, ainda não haviam sido implementadas, sendo necessário manter restrito o uso do aplicativo. Após a implementação dos itens acima citados, a configuração do aplicativo foi configurada para "Teste Alfa Aberto", onde qualquer pessoa com o link poderia realizar a instalação do aplicativo.

Ao abrir a aplicação, o usuário deverá criar um novo usuário ou utilizar uma combinação de *login* e senha pré-existente para obter acesso ao sistema. Uma vez feito o *login*, o usuário não necessitará inserir novamente suas credenciais, a não ser que o mesmo faça *logout*. Após autenticado com o servidor, pode-se utilizar as funcionalidades do aplicativo como criação de grupos, enviar convite de amizade para contatos, enviar convites para que os contatos se juntem grupos, adicionar

regiões conforme já explicado na seção de trabalhos relacionados, além de acompanhar outras pessoas de grupos em que o usuário já se encontra.

Figura 6 - Tela de testes do Follow Us



Fonte: Google (2016)

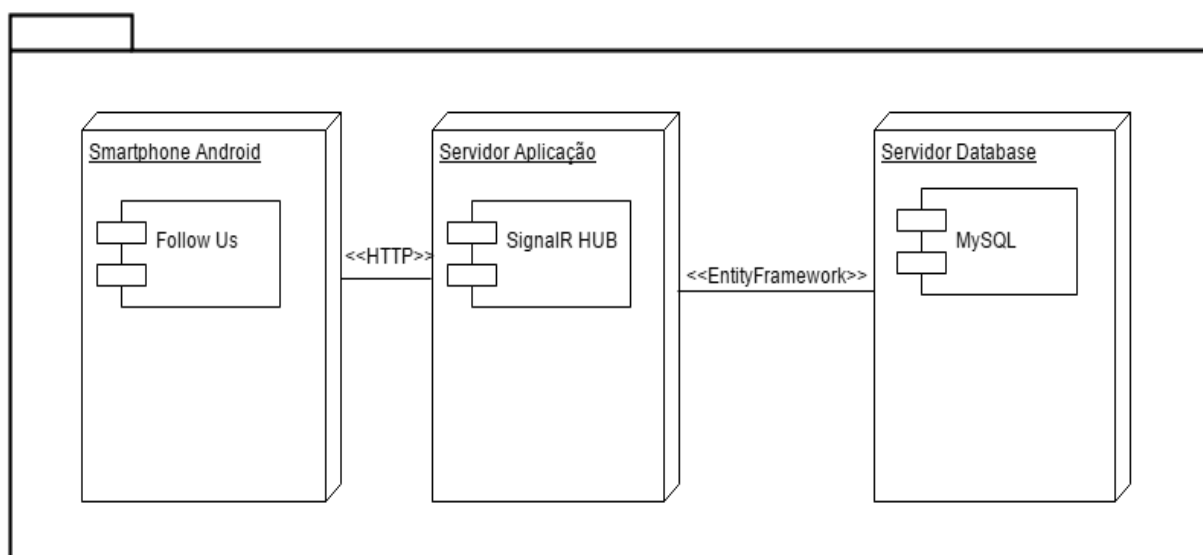
5.4 Arquitetura

De acordo com Garlan e Shaw (1994), a arquitetura de um software vai além dos algoritmos e estruturas de dados da computação. Conforme o autor, desenhar e

especificar a estrutura de um sistema emerge como um novo tipo de problema. As questões de estrutura incluem protocolos de comunicação, sincronização e acesso a dados, além de escalabilidade e desempenho. Complementando esta definição, de acordo com Bass, Clements e Kazman (2012), a arquitetura de software de um sistema, é uma estrutura de estruturas de um sistema, que compreende os elementos do software, as propriedades visíveis externamente destes elementos e os relacionamentos entre eles.

O aplicativo que resultou desta pesquisa, o Follow US, possui uma estrutura com componentes cuja função é bem definida. Para expor essa estrutura, foi elaborado um diagrama de implantação. De acordo com Martins (2007), este diagrama mostra como as várias partes físicas do sistema serão distribuídas no ambiente computacional e se comunicarão entre si, como por exemplo, máquinas cliente de usuários, servidores de banco de dados, servidores de comunicação e outros. Este diagrama também foi selecionado porque possibilita uma visão abrangente e de fácil entendimento da arquitetura.

Figura 7 - Diagrama de implantação

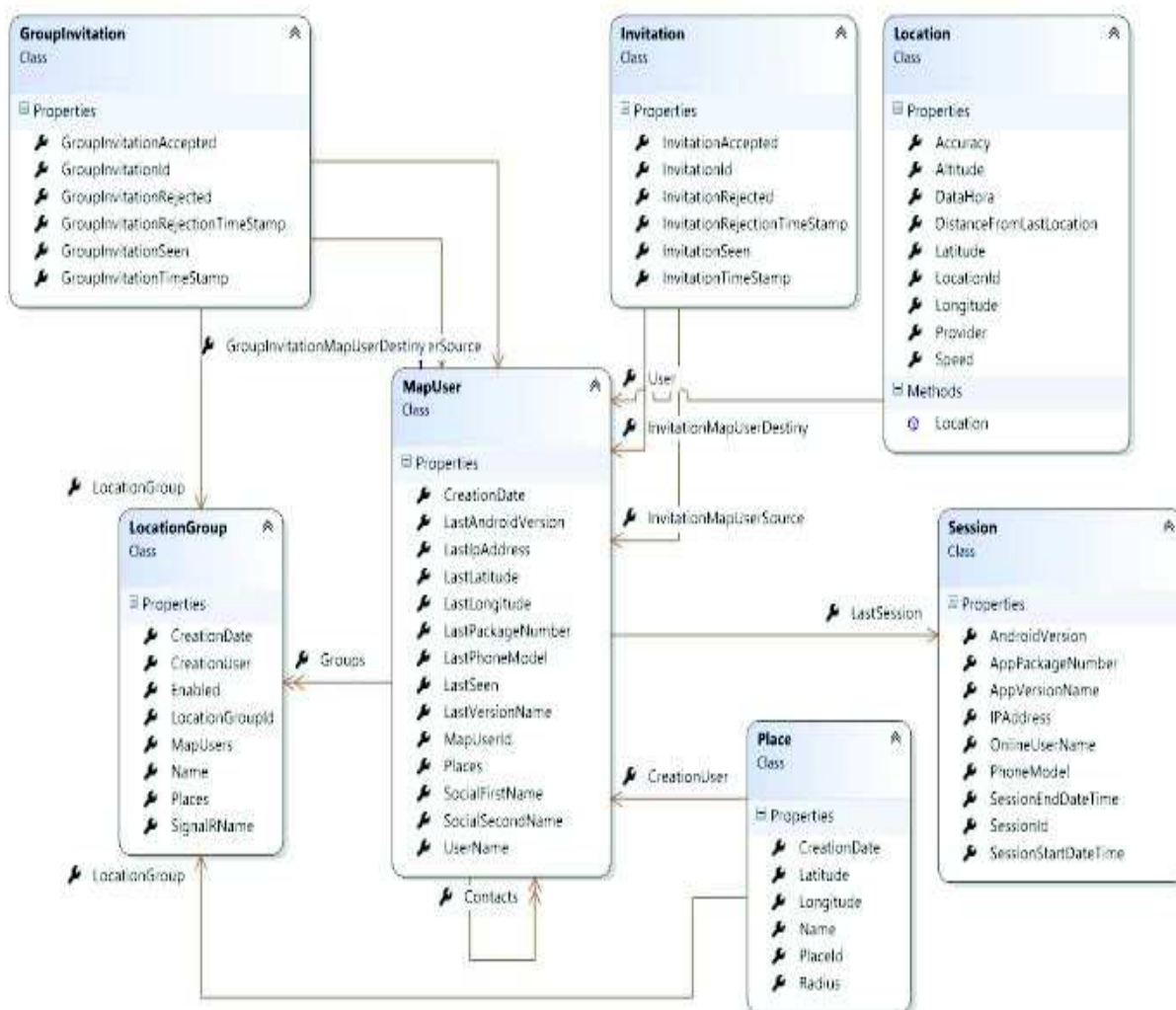


Fonte: Elaborado pelo autor

5.5 Diagrama de classes

De acordo com Fowler (2007), UML é uma família de notações gráficas apoiada por um metamodelo único que ajuda na descrição e no projeto de sistemas de software, particularmente daqueles construídos utilizando o estilo orientado a objetos. O diagrama é apresentado na figura 8.

Figura 8 - Diagrama de classes



Fonte: Elaborado pelo autor.

5.6 Tecnologias Utilizadas

5.6.1 Linguagem de Programação e IDE

A IDE escolhida para o desenvolvimento do projeto foi o *Visual Studio 2015*, da empresa Microsoft. Como a aplicação deve rodar na plataforma *Android*, foi então utilizada a extensão *Xamarin*, que permite que sejam desenvolvidos aplicativos para diversas plataformas móveis como *Android*, *iOS* e *Windows Phone* com uma mesma base de código na linguagem C# (*C Sharp*). Esta escolha foi feita devido ao fato de que a aplicação será mais útil e atingirá um maior público, se a mesma rodar em mais de uma plataforma. Com isto, diversos trechos código que contém chamadas a *Webservice*, persistência de dados, só para citar dois exemplos, podem ser reaproveitados se o desenvolvimento para *iOS* ou *Windows Phone* for iniciado.

5.6.2 Banco de Dados Local

O aplicativo tem como requisito não funcional o armazenamento de informações no dispositivo, possibilitando que informações como grupos e contatos sejam visualizadas mesmo em modo off-line. Para realizar esta tarefa, foi utilizado o banco de dados *SQLite* em conjunto com a biblioteca *SQLite.Net*, que é um *framework* ORM para persistência de dados para uso com o *SQLite*.

5.6.3 Servidor

Para hospedar os serviços que o aplicativo utiliza para autenticação e comunicação foi escolhido o IIS 8.0 da Microsoft, rodando uma aplicação ASP.NET. Esta aplicação ASP.NET possui um webservice exposto que por sua vez é utilizada para registro de usuário e autenticação. Também possui uma interface de conexão com o *SignalR*, framework selecionado para comunicação em tempo real.

5.6.4 Autenticação e Autorização

Para realizar a autenticação e autorização do usuário com o servidor, foi utilizado a tecnologia *ASP.NET Identity*. Este *framework* possibilita ao programador, de maneira prática, integrar na aplicação um serviço de autenticação e autorização. O *ASP.NET Identity* também faz o gerenciamento das *Access Tokens*, que são chaves geradas para confirmar a identidade de um usuário. Esta ferramenta também foi escolhida devido ao fato de que com ela, é possível integrar o processo de

autenticação usando as credencias de redes sociais como *Facebook*, *Twitter*, *Google Plus*, entre outras.

5.6.5 Banco de dados e Frameworks ORM

Para armazenar as informações da aplicação no servidor, foi escolhido o *MySQL*, um sistema gerenciador de banco de dados. Como o *Follow Us* se trata de um software desenvolvido sob o paradigma da orientação a objetos, foi selecionado como framework ORM o *Entity Framework* da Microsoft. Desta forma é possível trabalhar com objetos em memória além, que reduz a quantidade de linhas de código utilizadas realizando instruções SQL.

Também é importante citar aqui o processo de criação da estrutura do banco de dados. Conforme já citado no parágrafo anterior, a aplicação foi desenvolvida utilizando o paradigma da orientação a objetos. Por este motivo, primeiramente foram definidas as classes de negócio, isto é, as entidades e as relações entre elas. Após isto, foi utilizado o framework *Entity Framework Code First*, também da Microsoft, para gerar a estrutura do banco de dados. Esta tecnologia permite que a estrutura de um banco de dados seja definida a partir das entidades de um dado contexto.

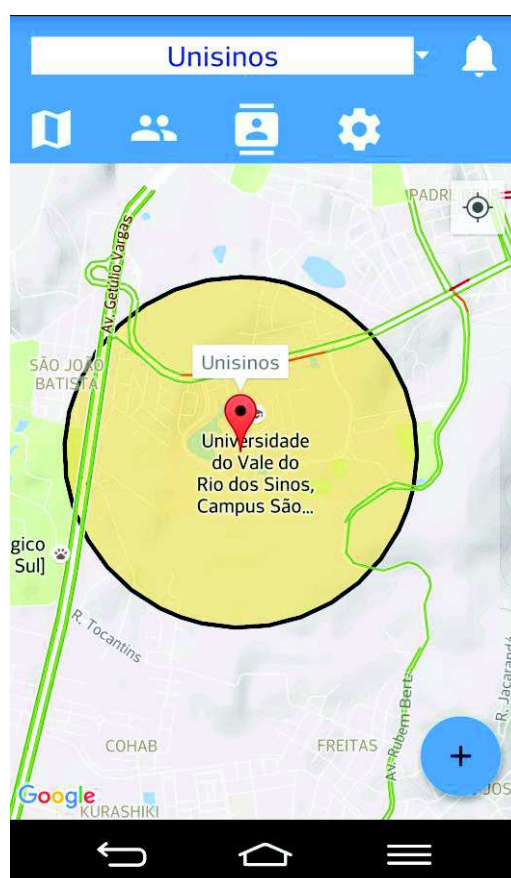
5.6.6 Framework para comunicação

Para realizar a comunicação entre cliente servidor foi utilizado o *framework SignalR*, já apresentado no sub capítulo 2.1. Conforme explicado, o *SignalR* possibilita que seja escolhido o protocolo ou arquitetura utilizada para enviar e receber informações. No caso deste trabalho foi selecionado o método *Server Sent Events* por proporcionar que uma conexão fique aberta entre cliente e servidor. É importante também citar o fato de que o *long-polling*, por abrir e fechar conexões constantemente, não proporciona uma forma eficiente para troca de dados em tempo real. Já os *WebSockets* não foram utilizadas neste caso pois conforme explicado, não é garantido que a conexão será estabelecida com sucesso em todos os casos, devido a bloqueios de conexões TCP/IP em *firewalls*.

5.6.7 Recursos para LBS

O Follow Us foi construído utilizando o conceito de Location Based Services, já apresentado no referencial teórico. Por este motivo se fez necessário a utilização de bibliotecas específicas para implementação das funcionalidades, como a exibição de mapas e de localização. Para a exibição dos integrantes dos grupos e cercas geográficas, foi utilizada a API *Google Maps V2* da empresa *Google*. Esta API oferece todas as ferramentas e funções necessárias para construção de aplicações baseadas em *Location Based Services*. Para obter a localização geográfica, foi utilizada a API nativa para uso do GPS do sistema operacional *Android*, que atendeu de forma satisfatória as necessidades para a construção do aplicativo.

Figura 9 – Mapa da API Google Maps V2 com geocerca



Fonte: Elaborada pelo autor

6 TESTES

6.1 Processo de testes

Na primeira fase de testes, foram selecionadas três pessoas que já haviam sido apresentadas a proposta inicial do aplicativo antes mesmo da primeira versão. Nos primeiros testes realizados, o aplicativo foi distribuído através de um link em servidor próprio. A bateria de testes iniciais envolveu a verificação de alguns fatores, como o desempenho das atualizações em tempo real, tanto em rede WiFi como em redes móveis, a precisão das localizações informadas pelo aplicativo, o comportamento do aplicativo com múltiplos usuários em movimento e a rapidez com que o aplicativo informa quando alguém entrou ou saiu de determinada região.

Inicialmente, o serviço que roda em segundo plano interagindo com o GPS, foi programado de forma a receber atualizações com um intervalo de tempo e distância inexistentes, ou seja, recebia atualizações com o máximo de frequência possível para o dispositivo, mesmo sem ter havido deslocamento. Nestes testes iniciais, verificou-se que tal frequência de atualizações esgotava rapidamente a bateria do dispositivo. Foram então feitas modificações no código fonte e novos testes. Na nova configuração, o dispositivo passou a receber atualizações do GPS a cada três segundos, além de a diferença de distância entre a localização anterior ter de ser de 7 metros. Após estas modificações, verificou-se grande melhora em relação a consumo de energia, além de não ter comprometido a precisão, das informações, tanto de localização quanto de velocidade, que o servidor recebia do dispositivo.

Pelo fato de as redes móveis serem imprevisíveis quanto a sua estabilidade, houve alguns problemas de queda de conexão por breves instantes. Foi necessário implementar um mecanismo de reconexão para que o aplicativo voltasse a enviar atualizações para outros usuários sem que fosse necessária intervenção humana.

Outro problema também encontrado foi o consumo excessivo de dados móveis. Durante testes verificou-se que o problema era na verdade o tamanho do JSON retornado pelo servidor ao cliente. A solução encontrada foi diminuir o

tamanho do nome dos campos, reduzindo o tamanho da informação. Nesta primeira fase, os problemas foram apenas documentados para posterior análise e correção no código-fonte.

Na segunda fase de testes, após a correção destes bugs que impediam o uso do aplicativo em uma situação real, o aplicativo foi submetido à loja de aplicativos do *Google*, a *PlayStore*. O aplicativo foi configurado para Testes Alfa, ou seja, qualquer um com o link disponibilizado pelo desenvolvedor pode fazer o download do mesmo. O referido link foi disponibilizado para o grupo de Escolta Olímpica da Polícia Rodoviária Federal que atuou nos Jogos Olímpicos no Rio de Janeiro. O aplicativo foi utilizado pela equipe para monitoramento das escoltas de delegações esportivas e autoridades de diversos países participantes das competições entre os locais de hospedagem e o parque olímpico. Outras atividades monitoradas através do Follow Us incluem as provas de Ciclismo de Estrada masculinas e femininas com largada na Praia do Pontal, escolta dos armamentos destinados ao tiro olímpico entre o porto do Rio de Janeiro e Deodoro e também escolta dos cavalos para as provas de hipismo. As duas últimas escoltas citadas foram de alto risco, pois além do alto valor dos bens transportados, as mesmas atravessaram regiões de conflito como a Avenida Brasil e Linha Vermelha. Nestas escoltas a equipe de APH (Atendimento pré-hospitalar) da Polícia Rodoviária Federal, denominada APH-3, monitorou o deslocamento da escolta, ficando baseada no bairro de Copacabana. Desta forma, caso ocorresse alguma emergência com a equipe de escolta, a equipe de APH poderia ser rapidamente acionada já com conhecimento do local da ocorrência. Vale ressaltar que a própria equipe de APH também poderia ser acompanhada, pela equipe de escolta através do aplicativo, caso necessário.

6.2 Resultados Obtidos

Nesta seção serão apresentados os resultados obtidos através das respostas dos participantes do teste do aplicativo no questionário. O teste foi constituído de oito participantes que avaliaram o aplicativo de acordo com o processo descrito no tópico anterior. O questionário possui seis perguntas, sendo que as três primeiras buscam saber a opinião do usuário em relação ao papel das funcionalidades do

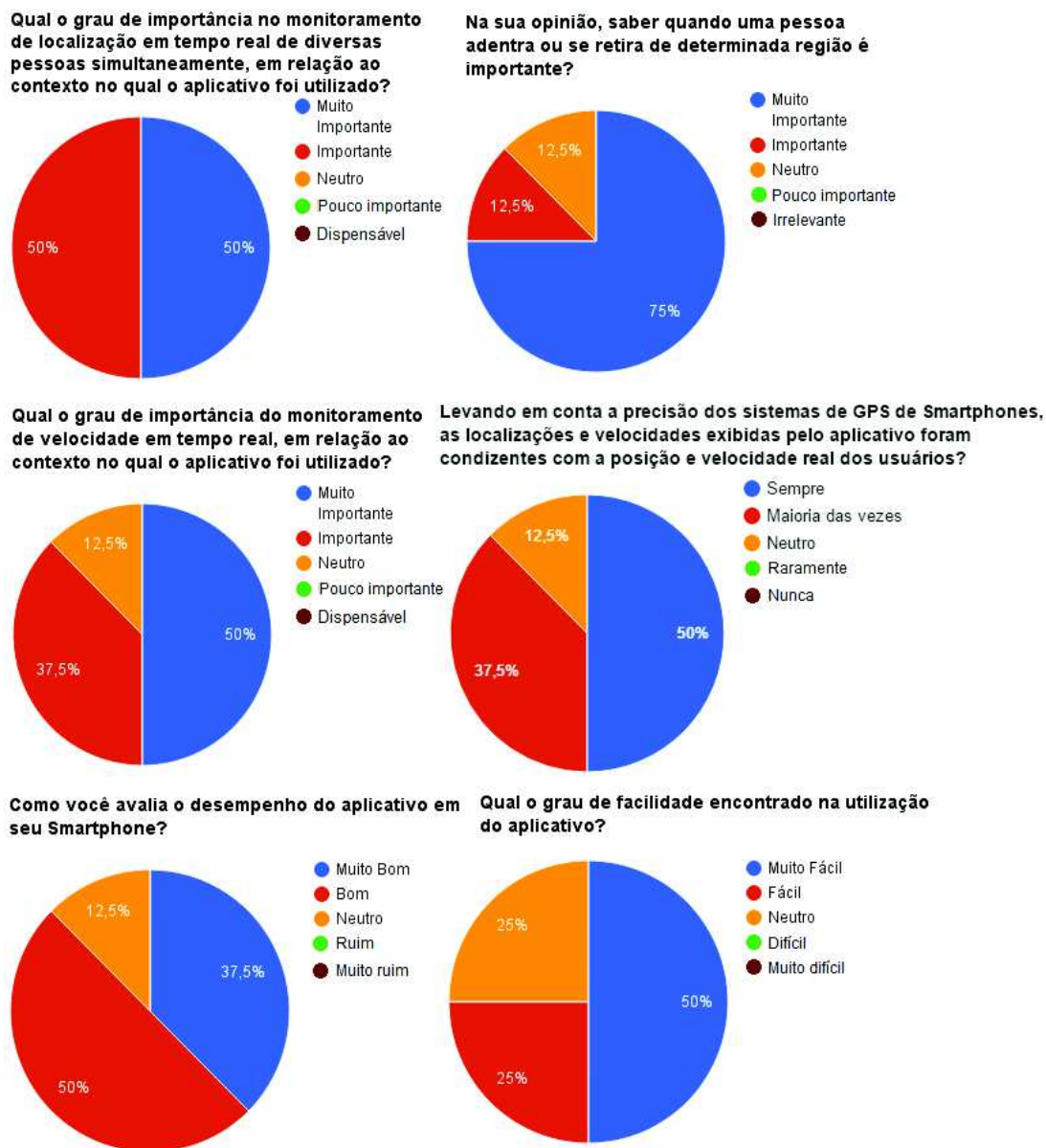
Follow Us. Já as três últimas tem como objetivo avaliar o desempenho, precisão e facilidade de uso do aplicativo, também na perspectiva do usuário. As respostas foram compiladas em um gráfico para cada pergunta, para facilitar o entendimento, conforme mostra a figura 10.

Como o Follow Us tem como principal funcionalidade o monitoramento de localização de um grupo de pessoas em tempo real, a primeira pergunta foi elaborada de forma a avaliar qual o grau de importância que este tipo de monitoramento tem na perspectiva do usuário levando em conta o contexto no qual o aplicativo foi utilizado. Como se pode observar no gráfico, todos os usuários consideram que o monitoramento de localização em tempo real é importante ou muito importante. A segunda pergunta busca saber a opinião dos usuários em relação ao uso de cercas geográficas, funcionalidade presente no Follow Us. Neste caso, os resultados indicam que 75% dos usuários consideram esta uma funcionalidade muito importante. A terceira pergunta é similar à primeira, porém tem como objetivo avaliar o grau de importância que o monitoramento de velocidade em tempo real possui em relação ao papel que o aplicativo desempenhou nos testes. Assim como na primeira pergunta, a maioria dos usuários também considera esta uma tarefa importante ou muito importante.

A quarta pergunta pretendia verificar se o aplicativo estava operando conforme planejado, exibindo as localizações e velocidades corretas. Como o aplicativo foi testado por uma equipe de escolta, torna-se fácil saber se a localização, assim como a velocidade, reflete a realidade. O resultado obtido com esta questão indica que sete entre os oito participantes obtiveram informações consistentes na maioria das vezes.

A quinta questão buscou avaliar o desempenho do aplicativo no Smartphone dos testadores na perspectiva do usuário. Conforme o resultado, 87,5% dos participantes considera o desempenho bom ou muito bom. Já a última pergunta buscou avaliar qual o grau de facilidade encontrado na utilização do aplicativo, cujo resultado indica que 75% dos usuários encontraram facilidade ao utilizar a aplicação.

Figura 10 - Resultados do questionário



Fonte: Elaborado pelo autor

7 CONCLUSÃO

É perceptível que o compartilhamento de localização e velocidade em tempo real é algo que pode contribuir de forma considerável para o desenvolvimento de

atividades que envolvam deslocamento e coordenação entre pessoas a partir de diferentes localizações.

Conforme demonstrado no capítulo anterior, o Follow Us se apresentou, sob a ótica do usuário final, como um aplicativo capaz de proporcionar este recurso de forma satisfatória. As funcionalidades oferecidas pelo aplicativo foram plenamente utilizadas pelos usuários durante o processo de testes, tendo sido recebidas com grande aceitação pelos mesmos, como se pode observar nos resultados obtidos com o questionário da pesquisa.

Apesar de grande aceitação entre os usuários, verificou-se que no tocante a facilidade de uso e desempenho do aplicativo, existe oportunidades de melhorias. Mesmo que a maioria dos usuários tenha encontrado facilidade na utilização do Follow Us, é importante frisar que foi necessário realizar uma breve explicação sobre o funcionamento do aplicativo aos participantes do teste, principalmente na regra de negócio relacionada ao processo de adicionar pessoas ao grupo. Uma breve introdução e apresentação das funcionalidades, assim como encontrado em diversos aplicativos disponíveis na *Play Store*, seria de grande utilidade neste caso.

No que diz respeito a questão do desempenho da aplicação, pode-se melhorar o mecanismo de sincronia dos dados do servidor com o banco de dados local da aplicação. O aplicativo utiliza as informações salvas no dispositivo para exibi-las ao usuário, para que este consiga visualizar as últimas informações obtidas, mesmo estando sem conexão, possibilitando assim uma melhor experiência. No entanto, quando o aplicativo se conecta com servidor, as informações armazenadas no servidor precisam ser obtidas para que seja feita uma comparação entre os dados existentes localmente e remotamente, e com isso a sincronização dos dados. Uma notável melhoria no consumo de dados móveis adviria de um controle de sincronização no servidor, possibilitando que somente novas informações fossem enviadas ao aplicativo, e indiretamente influenciando o uso de poder computacional, que seria diminuído, pelo menos em partes, na rotina de sincronização.

Por fim, este projeto oferece diversas direções para novas pesquisas que visem adicionar novos recursos e melhorar os já existentes. A expansão dos

recursos relacionados à rede social como a criação de grupos e adicionar amigos à sua lista de contatos mostra um enorme potencial, principalmente quando existe a possibilidade de integrar o aplicativo com recursos de outras redes sociais como o *Facebook*. Este projeto não teve foco na segurança da informação, não abordando a implementação de mecanismos de criptografia, sendo esta, portanto, outra perspectiva em cima da ideia inicial do Follow Us.

REFERÊNCIAS

BASS, L.; CLEMENTS, P.; KAZMAN, R. **Software Architecture in Practice**. 3. ed. Indianapolis: Addison-Wesley Professional, 2012.

CHEN, R; GUINNESS, R. E; **Geospatial computing in mobile devices**; p. 140; Norwood: Artech House, 2014;

EL-RABBANY, A; **Introduction to GPS: The Global Positioning System**; Mobile communication series; p. 1; Norwood: Artech House, 2002.

FLETCHER, P. **Introduction to SignalR**. 2014. Disponível em: <<http://www.asp.net/signalr/overview/getting-started/introduction-to-signalr>>. Acesso em: 20 jun. 2016.

FOWLER, M. **UML Essencial: Um Breve Guia para Linguagem Padrão**. 3ª edição. p. 25-52. São Paulo: Bookman, 2007.

GARLAN, D; SHAW, M; **An Introduction to Software Architecture, Advances in Software Engineering and Knowledge Engineering**, p. 2. Volume I, World Scientific. Pittsburgh: 1993;

GRESSLER, A. L; **Introdução à pesquisa: Projetos e relatórios**. P. 43, São Paulo: Loyola, 2004.

GOOGLE. **Teste de App para Android**. Mountain View, 2016. Disponível em:<<https://play.google.com/apps/testing/com.gpsapp.followus>>. Acesso em: 7 jul. 2016.

LIMA, M. C; **Monografia – A engenharia da produção acadêmica**. São Paulo: Saraiva, 2004.

MARTINS, J. C. C. **Técnicas para Gerenciamento de Projetos de Software**. p. 208. Rio de Janeiro: Brasport, 2007.

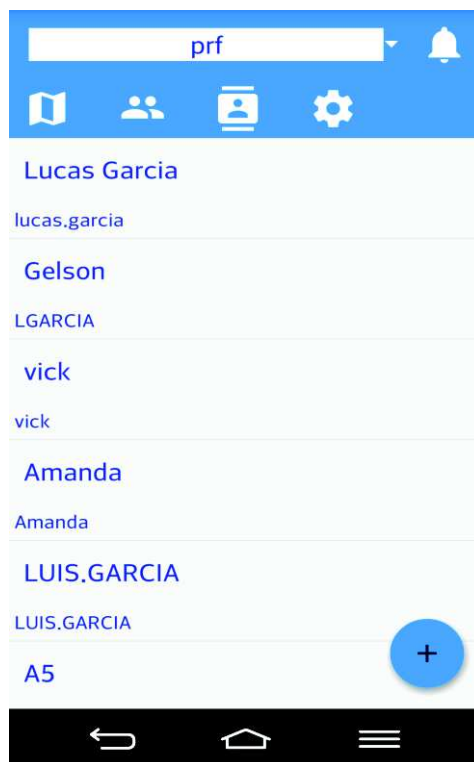
MATERA, M; ROSSI, G; **Trends in Mobile Web Information Systems.** p. 137.
Cham: Springer, 2013.

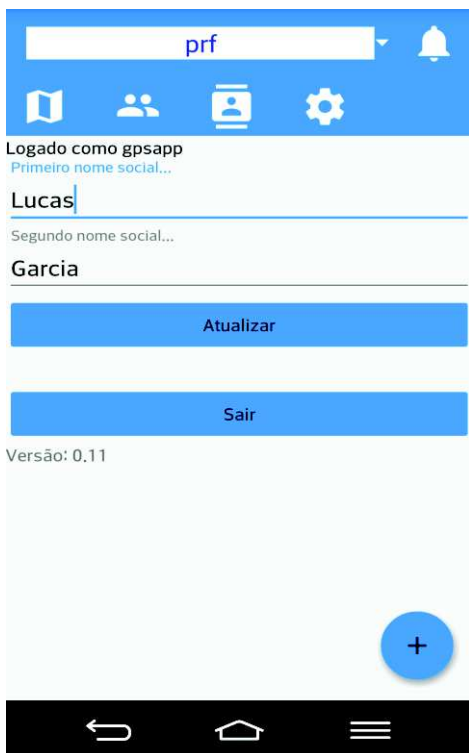
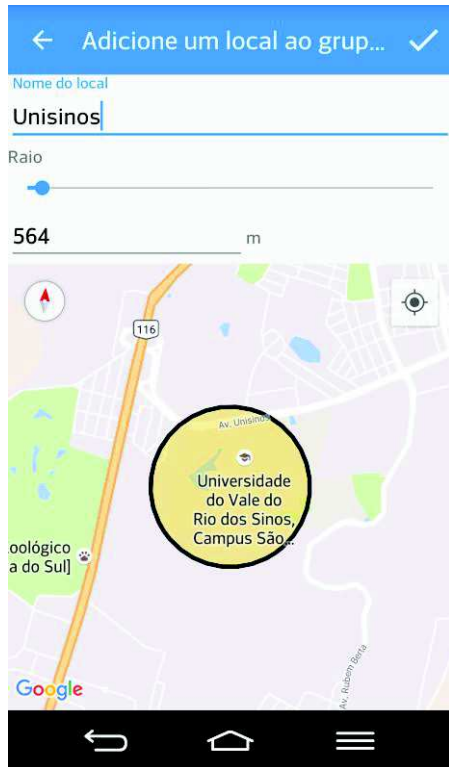
SCHILER, J; VOISARD, A; **Location-Based Services.** p.1,10 San Francisco:
Elsevier, 2004.

WILSON, M. **Location-based services, conspicuous mobility, and the location-aware future,** 2012.

APÊNDICE A – TELAS DO APLICATIVO







APÊNDICE B – FORMULÁRIO DO QUESTIONÁRIO

Avaliação do Follow Us

Este questionário foi elaborado com o intuito de avaliar o desempenho e usabilidade do Follow Us, na perspectiva do usuário. A sua participação é anônima, portanto sua identidade não será revelada, tampouco suas respostas. O resultado deste questionário será apresentado no artigo "Follow Us: Aplicativo para compartilhamento de localização em tempo real", que é pré-requisito para a conclusão do curso de Especialização em Desenvolvimento de Aplicações para Dispositivos Móveis, na instituição de ensino Unisinos.

* Required

Qual o grau de importância no monitoramento de localização em tempo real de diversas pessoas simultaneamente, em relação ao contexto no qual o aplicativo foi utilizado? *

	1	2	3	4	5	
Dispensável	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito importante

Na sua opinião, saber quando uma pessoa adentra ou se retira de determinada região é importante? *

	1	2	3	4	5	
Inrelevante	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito importante

Qual o grau de importância do monitoramento de velocidade em tempo real, em relação ao contexto no qual o aplicativo foi utilizado? *

	1	2	3	4	5	
Dispensável	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Muito importante

Levando em conta a precisão dos sistemas de GPS de Smartphones, as localizações e velocidades exibidas pelo aplicativo foram condizentes com a posição e velocidade real dos usuários? *

	1	2	3	4	5	
Nunca	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Sempre

Como você avalia o desempenho do aplicativo em seu Smartphone? *

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

Qual o grau de facilidade encontrado na utilização do aplicativo? *

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