

**SISTEMA COLABORATIVO PARA ORIENTAÇÃO DE AGENTES DE TRÂNSITO**Vinicius Orestes Alves<sup>1</sup>Prof. Msc. Daniel Formolo<sup>2</sup>

Resumo: O presente artigo descreve a construção de um sistema colaborativo para orientação de agentes de trânsito. O objetivo do sistema é auxiliar o cumprimento das leis de trânsito, bem como guiar os agentes para os lugares chave da cidade onde o trânsito encontra-se mais congestionado. Qualquer pessoa pode contribuir com informações sobre as infrações que estão acontecendo, os lugares onde elas mais acontecem ou simplesmente problemas no trânsito devido a, por exemplo: alto fluxo de automóveis, sinalizadas estragadas, acidentes, etc. As colaborações são salvas no sistema e ficam disponíveis para serem consultadas por agentes de fiscalização. Dessa forma, os agentes podem monitorar em tempo real os problemas e infrações que estão acontecendo na cidade e agir com maior rapidez e inteligência. O cidadão contribui com informações importantes para o bem geral da sociedade, auxiliando a boa manutenção do trânsito e ampliando o sentimento de cidadania. O artigo detalha os desafios encontrados na construção de um sistema colaborativo, as soluções adotadas, bem como os resultados atingidos.

Palavras-chave: Sistemas colaborativos. Dispositivos móveis. Android.

**1 INTRODUÇÃO**

Fiscalização é a prática de vigilância constante sobre determinada atividade que tenha seu procedimento regulado por lei específica. Com o objetivo de fazer-se cumprir as leis, a fiscalização deve ser constante. Somente os agentes de trânsito são capazes de aplicar as devidas punições que decorrem do descumprir da lei, mas muitas vezes os agentes sozinhos não são capazes de fiscalizar grandes áreas por longos períodos de tempo.

Com um sistema colaborativo para orientação de agentes de trânsito, os cidadãos podem contribuir com a fiscalização das leis. As pessoas contribuem não diretamente aplicando as devidas punições, mas sim alertando os agentes sobre os

---

<sup>1</sup> Vinicius Orestes Alves, graduado em Ciência da Computação pela ULBRA. Atualmente cursando Especialização em Desenvolvimento de Aplicações para Dispositivos Móveis na Unisinos <vinicius.voa@gmail.com>.

<sup>2</sup> Daniel Formolo, professor no curso de Especialização em Desenvolvimento de Aplicações para Dispositivos Móveis <danielformolo@unisinos.br>.

problemas que estão acontecendo ou que acabaram de acontecer. Dessa forma os agentes se deslocam para o local, avaliam a situação e atuam conforme a necessidade. As pessoas também podem contribuir com informações gerais sobre os locais onde o trânsito está com problemas devido a, por exemplo: alto fluxo de automóveis, sinalizadas estragadas, acidentes e outros motivos que requeiram uma orientação especial dos agentes de trânsito.

## 2 APLICATIVOS COLABORATIVOS

Quando se pensa em aplicativos colaborativos, deve-se imaginar cada usuário como um colaborador do sistema. Para o sucesso do sistema, deve-se pensar em aplicações que motivem a colaboração das pessoas, em outras palavras, se as pessoas não estiverem interessadas em colaborar com o sistema, ele tende ao fracasso.

Um termo muito comum na área de aplicativos colaborativos é *crowdsourcing*. Este termo foi definido por Jeff Howe e Mark Robinson em Junho de 2006 na revista *Wired* como sendo um novo modelo de negócio que aproveita as soluções criativas de uma rede distribuída de pessoas. *Crowdsourcing* também pode ser considerado como o ato de uma empresa ou instituição tornar uma função que era feita por funcionários em uma função terceirizada para uma rede indefinida (e geralmente grande) de pessoas (BRABHAM, 2008).

Atualmente existem muitos casos de aplicativos colaborativos de sucesso. Os GPSs sociais são bons exemplos desses aplicativos colaborativos que deram certo. As pessoas colaboram com informações em tempo real sobre trânsito, de maneira que o GPS sabe informar a rota mais inteligente. Dessa forma o aplicativo considera não apenas a rota com a distância mais curta, mas sim a melhor rota. Ele considera entre os vários fatores que afetam o deslocamento, a rota com melhor fluxo de automóveis para aquele exato momento. Essas informações sobre o trânsito são fornecidas de forma colaborativa pelos usuários do aplicativo.

Um exemplo de GPS colaborativo é o *Waze*. Trata-se de um GPS *turn-by-turn*, ou seja, um sistema de navegação no qual as direções para a rota selecionada são continuamente apresentadas ao usuário na forma de instruções de voz e visual

(BUTTON, 2001). Existem versões do aplicativo para Android e iOS. O aplicativo conta com uma série de funcionalidades, sendo que a principal delas é baseada no sistema colaborativo.

Os usuários do *Waze* podem fornecer informações sobre o trânsito, incluindo: acidentes, buracos, obras, congestionamentos, etc. Assim, quando for elaborar o trajeto, o usuário é avisado pelo *Waze* sobre eventuais problemas que pode encontrar no caminho.

O *Waze* oferece diversas alternativas de caminhos menos problemáticos e também pode recalcular a rota sempre que necessário. Um problema desse aplicativo reside no fato de ele não ser muito utilizado no Brasil. Nos grandes centros a presença dos usuários é maior, mas longe deles, não (INFO, 2013). A Figura 1 apresenta algumas telas do aplicativo colaborativo *Waze*.



Figura 1: *Waze*.

Outro exemplo de aplicativo colaborativo é o *SocialFuel*. Trata-se de um aplicativo que pode ajudar os motoristas que deixaram o tanque do carro esvaziar. Ele localiza os postos de combustíveis mais próximos e mostra o endereço de cada um. O aplicativo faz isso utilizando o sistema de GPS do *smartphone* ou *tablet*.

Um problema do *SocialFuel* é que ele não conta com uma ferramenta de pesquisa por endereços, ou seja, ele mostra apenas os postos de combustível que estão mais próximos. Além disso, muitas vezes os valores da gasolina, etanol, diesel e do GNV estão desatualizados. Dessa forma o aplicativo é realmente útil apenas para informar onde estão os estabelecimentos (INFO, 2013).

O fato do *SocialFuel* não manter atualizados os valores cobrados pelos postos de combustíveis é um claro exemplo de problema que pode-se resolver com o aumento do número de usuários do software colaborativo. Quanto maior o número de pessoas usando o software colaborativo, melhores os resultados, pois existem mais pessoas contribuindo e confirmando as informações que já existem no sistema. Por sua vez o aplicativo também se torna mais atraente ao público por oferecer um serviço de melhor qualidade. A Figura 2 apresenta uma das telas do aplicativo colaborativo *SocialFuel*.



Figura 2: *SocialFuel*.

O objetivo desse trabalho não é desenvolver um GPS colaborativo, mas sim um aplicativo que, com base nas informações fornecidas pelas pessoas, auxilie os agentes de trânsito no trabalho de fiscalização e organização do trânsito. As informações fornecidas pelas pessoas devem ajudar os agentes a se deslocarem mais inteligentemente para os locais onde estão ocorrendo problemas ou infrações de trânsito naquele exato momento.

Além de ajudar os agentes a atuar de forma mais eficiente, os dados coletados pelo sistema colaborativo podem ser analisados de maneira a se extrair informações sobre o comportamento do trânsito em determinada cidade. Isso pode auxiliar na definição de medidas preventivas como troca de sentido de vias e deslocamento das equipes com antecedência, nos horários corretos, para os pontos com maior incidência de problemas.

### **3 DESENVOLVIMENTO DO APLICATIVO**

Devido ao alto número de usuários (que podem vir a se tornar colaboradores do sistema) decidiu-se inicialmente criar um aplicativo para a plataforma Android. Obtendo sucesso de uso do aplicativo colaborativo nessa plataforma, posteriormente pensa-se no desenvolvimento para outras plataformas. Nesse caso, o desenvolvimento para outras plataformas é mais simples e rápido. O aplicativo colaborativo deve tentar atingir o maior número possível de plataformas com o objetivo de atrair mais colaboradores.

Existem diversas versões de Android no mercado. Ainda visando abranger o maior número possível de colaboradores optou-se por desenvolver um aplicativo de alta compatibilidade. Definiu-se respectivamente como versão mínima e máxima as APIs do Android level *10* e *17*. Dessa forma o aplicativo torna-se compatível com dispositivos móveis que têm versões de Android entre 2.3.3 e 4.2.2 (ANDROID DEVELOPER GUIDE, 2013).

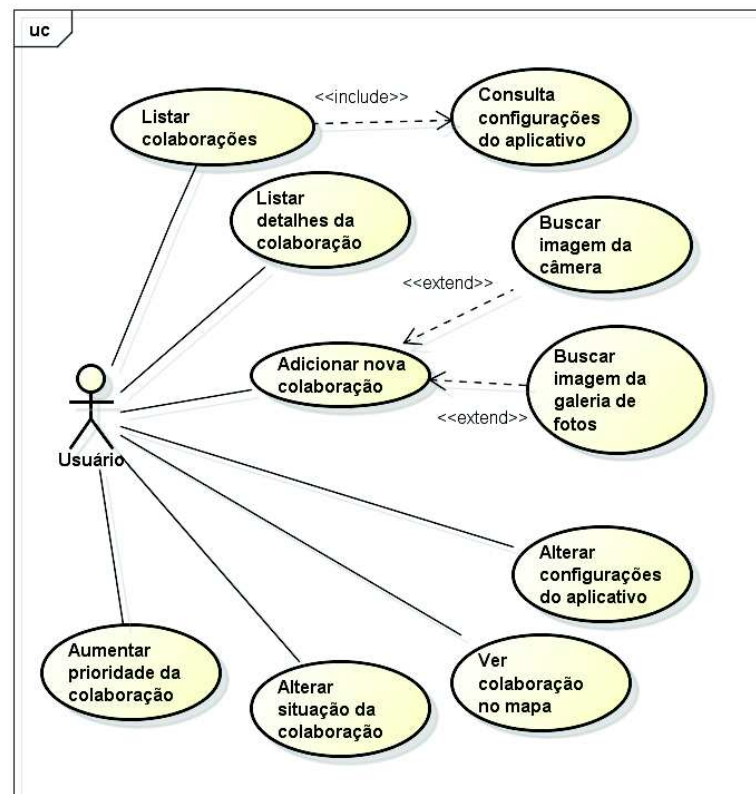
Os principais fluxos de navegação entre as telas do aplicativo podem ser encontrados no Anexo A. Todas as imagens foram obtidas de um dispositivo móvel com sistema operacional Android versão 2.3.6.

### 3.1 Casos de uso do aplicativo

Para esta versão inicial do aplicativo, foram desenvolvidos apenas os casos de uso mais básicos. Os casos de uso desenvolvidos são:

- 1) Usuário abre o aplicativo e tem acesso a uma lista das colaborações.
- 2) Usuário altera as configurações do aplicativo.
- 3) Usuário adiciona uma nova colaboração.
- 4) Usuário visualiza os detalhes de uma colaboração.
- 5) Usuário altera a situação de uma colaboração.
- 6) Usuário aumenta a prioridade de uma colaboração.
- 7) Usuário visualiza uma colaboração no mapa.

Esses casos de uso são ilustrados abaixo no diagrama da Figura 3. O diagrama considera apenas os cenários básicos para essa versão inicial. Não são considerados os possíveis casos de uso futuros que podem conter, por exemplo, usuários com permissões específicas de agente de trânsito.



powered by Astah

Figura 3: Diagrama de Caso de Uso do Aplicativo.



### 3.1.1 Listar colaborações

Ao abrir o aplicativo a posição do usuário é obtida com o uso do GPS. Caso o GPS esteja desabilitado, o usuário recebe uma mensagem informativa explicando que o GPS deve ser habilitado e o aplicativo é encerrado. Caso a posição seja obtida com sucesso o usuário visualiza a lista das colaborações. A lista considera ordenação e filtros previamente configurados e que podem ser alterados a qualquer momento em uma tela específica de configurações do aplicativo.

Os atributos de cada item da lista são: o título, a data de envio, a distância em relação à posição atual e por fim a prioridade da colaboração. A prioridade é apresentada ao lado de um ícone do Android. O humor do ícone muda de acordo com a situação atual da colaboração: triste para colaborações novas, um pouco menos triste para colaborações em atendimento e feliz para colaborações resolvidas.

Para calcular a distância entre as colaborações que já estão cadastradas no sistema e a posição atual do colaborador, utilizou-se a fórmula de Haversine. Trata-se de uma equação usada em navegação que fornece a distância entre dois pontos usando latitudes e longitudes (WIKIPEDIA, 2013).

Pode-se saber a situação da colaboração através da cor do título apresentado. As possíveis situações das colaborações no sistema são: “Nova”, “Em atendimento” e “Resolvida” a representação é feita respectivamente através das cores “Vermelho”, “Amarelo” e “Verde”.

Para os casos nos quais fotos foram enviadas no momento em que a colaboração foi adicionada, é possível visualizar (em tamanho pequeno) uma prévia da imagem que será apresentada no momento que a colaboração é detalhada. Para os casos onde fotos não foram enviadas é sempre exibida uma imagem padrão.

Navegando até o final da lista encontra-se um botão para carregar mais colaborações. Toda vez que esse botão é acionado o tamanho da lista é dobrado e a posição da barra de rolagem é mantida. No caso do tamanho da lista ser mantido, o usuário recebe uma mensagem informando que todas as colaborações para as preferências atuais do aplicativo já foram carregadas. Também existe um filtro no

topo da lista que pode ser usado para filtrar as colaborações pelo título. O filtro é aplicado apenas nas colaborações que já estão carregadas na lista.

A Figura 4 apresenta a tela principal do aplicativo que é a lista de colaborações. Na esquerda a tela é apresentada na sua forma normal e na direita com o menu do Android acionado.



**Acionando o menu do Android**

Figura 4: Listar colaborações.

O aplicativo faz uso da biblioteca "*Universal Image Loader*" para carregar e apresentar as imagens que aparecem na lista de colaborações e também quando a colaboração é detalhada. O uso da biblioteca permite obter um melhor desempenho fazendo *cache* das imagens no disco.

Com o uso da biblioteca "*Universal Image Loader*" também foi possível mostrar as imagens com cantos arredondados na lista de colaborações. A biblioteca também permitiu definir uma imagem temporária que é mostrada enquanto a imagem desejada esta sendo carregada. Também foi possível definir uma imagem que deve ser apresentada caso ocorra algum problema no carregamento da imagem desejada (UIL, 2013).



### 3.1.2 Alterar configurações

O usuário pode alterar as configurações do aplicativo com o objetivo de alterar a ordenação e filtros do conteúdo que é apresentado na lista de colaborações. Acessando-se o menu a partir da lista de colaborações existe a opção de acessar a tela de configurações do aplicativo.

A tela de configurações permite definir como a lista será ordenada. A lista pode ser ordenada de forma a mostrar mais ao topo as colaborações: com maior prioridade, mais recentes ou as mais próximas da posição atual do usuário.

A ordenação pelas colaborações mais próximas é um caso de uso interessante principalmente para futuras versões do aplicativo nas quais poderia existir um usuário agente de trânsito que poderia buscar as ocorrências mais próximas da sua posição atual e assim se deslocar com mais inteligência. A tela também permite filtrar as colaborações pelas seguintes situações: “Nova”, “Em atendimento” e “Resolvida”.

Definiu-se que por padrão as colaborações são carregadas sem nenhum filtro, sendo que as mais recentes são mostradas mais ao topo da lista. As configurações podem ser alteradas a qualquer momento e são sempre lembradas, mesmo quando o aplicativo é reiniciado.

As configurações do aplicativo são mantidas usando “*shared preferences*”. Esse é um recurso do Android que permite salvar e recuperar os dados a qualquer momento, mesmo depois de encerrar o aplicativo (ANDROID DEVELOPER GUIDE, 2013). A Figura 5 mostra a tela de configurações.

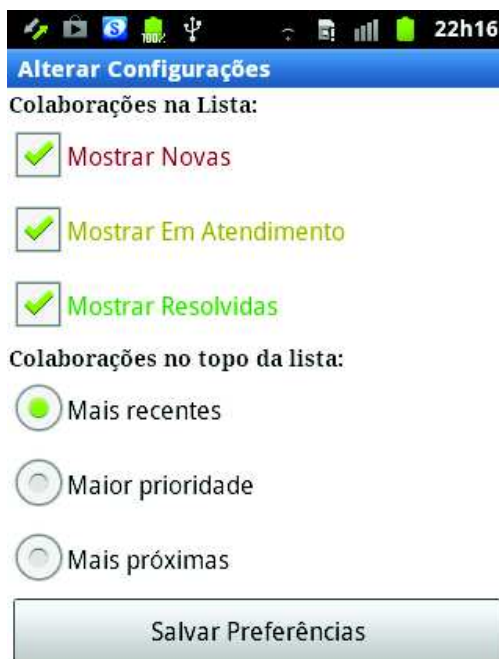


Figura 5: Tela de configurações.

### 3.1.3 Adicionar colaboração

Ao adicionar uma nova colaboração o usuário informa o título e também uma breve descrição. O campo título sugere possíveis exemplos, o que facilita o cadastro evitando excesso de digitação.

A sugestão de título também direciona os usuários a manterem um padrão para títulos, o que facilita a geração de relatórios no caso de uma possível integração com algum sistema integrado ao servidor que recebe as mensagens dos dispositivos móveis.

A localização é obtida automaticamente usando o GPS. Opcionalmente também existe a possibilidade de adicionar uma foto do local. A foto pode ser obtida diretamente com a câmera ou pode ser obtida da galeria de fotos do Android.

A foto traz credibilidade para a colaboração. O envio da foto requer maior comunicação com o servidor, ou seja, enviam-se mais dados. O usuário tem a opção de colaborar sem o envio da foto, economizando banda na transferência de dados, porém essa colaboração possivelmente terá menor credibilidade no sistema. Os dados são codificados em base 64 para serem enviados ao servidor. A data de envio

da colaboração é obtida automaticamente do sistema operacional. A Figura 6 ilustra a tela responsável por adicionar uma nova colaboração.



Figura 6: Adicionar colaboração.

Ao clicar na imagem da câmera, que pode ser vista a esquerda na Figura 6, são apresentadas as opções para obter a foto da câmera ou da galeria do Android. Esta seleção pode ser vista a direita na Figura 6. Depois que a foto é obtida ela é apresentada no lugar da imagem da câmera e o texto “Selecionar Imagem Opcional” desaparece.

#### 3.1.4 Visualizar os detalhes de uma colaboração

O usuário tem a opção de visualizar os detalhes de uma colaboração. Ao clicar-se em uma colaboração na lista de colaborações, são apresentados os detalhes da colaboração.

Além de apresentar os detalhes da colaboração, são disponibilizadas opções no menu para: alterar a situação da colaboração, aumentar a prioridade ou até mesmo visualizar o local da colaboração no mapa.

A Figura 7 apresenta a tela de detalhes da colaboração. Na esquerda a tela é apresentada na sua forma normal e na direita com o menu do Android acionado.



Figura 7: Detalhes da colaboração.

### 3.1.5 Alterar a situação de uma colaboração

O usuário pode alterar a situação de uma colaboração. Para essa versão inicial qualquer usuário pode alterar a situação da colaboração para “Em atendimento” ou “Resolvida”, portanto conta-se com a boa fé dos usuários.

Para versões futuras, será agregado ao sistema um cadastro de usuários, então apenas usuários com a permissão adequada poderão alterar situação das colaborações. Os usuários que terão essa permissão serão os agentes de trânsito. Alterar a situação de uma colaboração é uma opção do menu da tela de detalhes da colaboração. Ao clicar nessa opção o resultado é o apresentado na Figura 8.



Figura 8: Alterar situação da colaboração.

### 3.1.6 Aumentar a prioridade de uma colaboração

O usuário pode considerar que alguma questão é crítica e aumentar a prioridade. Esse aumento de prioridade ajuda a colaboração a ser apresentada mais ao topo da lista (se a lista estiver ordenada por prioridade).

O aumento de prioridade também serve como confirmação de que a colaboração é realmente verdadeira. Essa confirmação decorre do fato de que varias pessoas estão confirmando que o problema esta realmente acontecendo e causando transtornos para a sociedade.

Os agentes devem considerar um atendimento prioritário para as colaborações que têm maior prioridade com o objetivo de atender as maiores necessidades dos colaboradores. Aumentar a prioridade de uma colaboração é uma opção do menu da tela de detalhes da colaboração. Essa opção é destacada na Figura 9.



Figura 9: Aumentar prioridade da colaboração.

### 3.1.7 Visualizar colaboração no mapa

Quando a colaboração é adicionada, a posição geográfica do usuário é obtida automaticamente com o uso do GPS. Caso o usuário queira visualizar a colaboração no mapa, as informações de latitude e longitude da colaboração são usadas para marcar um ponto no mapa.

Foi utilizado Google Maps API V2 para desenhar o mapa e marcar o ponto correspondente à colaboração (ANDROID DEVELOPER GUIDE, 2013). O ponto é marcado no mapa usando o ícone do aplicativo. Visualizar a colaboração no mapa é uma opção do menu da tela de detalhes da colaboração. Ao clicar nessa opção o resultado é o apresentado na Figura 10.





Figura 10: Visualizar colaboração no mapa.

A API V2 do Google Maps requer que o dispositivo móvel tenha o Google Play Services instalado. O aplicativo deve ser apropriadamente autorizado a acessar a API. Isso requer uma sequência de passos bem definidos na documentação oficial do Google Maps (ANDROID DEVELOPER GUIDE, 2013).

### 3.2 Principais classes e suas funções

Com o objetivo de manter a organização do projeto, as classes do aplicativo foram divididas em três pacotes. O pacote principal contém as classes que representam as telas do aplicativo, ou seja, as *Activities*. O pacote *model* possui as classes que representam o modelo de dados. O pacote útil possui classes que são úteis ao projeto como um *todo*. A organização das classes em pacotes é apresentada na Figura 11.

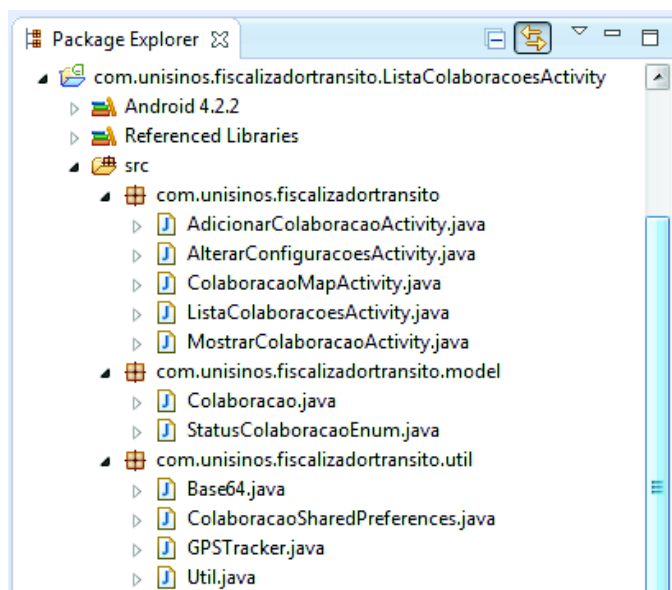


Figura 11: Organização das classes em pacotes.

No geral o nome das classes é autoexplicativo e representa a utilidade da classe no projeto. Cada tela do aplicativo é uma *Activity*. As classes com o sufixo “*Activity*” representam as telas.

No pacote *model* encontra-se a classe “*Colaboracao*” que é responsável por representar uma colaboração no sistema. Também no pacote *model* encontra-se uma classe de enumeração chamada “*StatusColaboracaoEnum*” que representa todas as situações possíveis nas quais uma colaboração pode ser encontrada no sistema.

O pacote útil possui diversas classes com métodos úteis ao projeto como um todo. Entre essas utilidades podem-se destacar métodos para codificar fotos para o formato base 64, métodos para trabalhar com o GPS, métodos para trabalhar com *shared preferences*, métodos para validação e formatação de dados, etc. As principais classes do aplicativo podem ser encontradas no diagrama de classes do Anexo B.

### 3.3 Desenvolvimento dos serviços

O aplicativo consulta, cria e atualiza os dados das colaborações. Dessa forma, fez-se necessário a construção de alguns serviços web para suportar essas operações.

Os serviços foram desenvolvidos na linguagem PHP. Eles acessam um banco de dados MySQL com o objetivo de consultar, criar e atualizar colaborações. Eles foram testados em um servidor web Apache.

Para esta versão inicial do aplicativo, não foi desenvolvido nenhum sistema de autenticação, portanto os serviços estão abertos e totalmente vulneráveis a possíveis ataques. Essas vulnerabilidades devem ser resolvidas antes de uma possível publicação da aplicação.

No banco de dados MySQL criou-se uma base de dados exclusiva para o aplicativo. Em seguida a tabela “colaborações” foi criada para armazenar todas as colaborações. A Figura 12 ilustra a tabela e seus respectivos atributos.

**colaboracoes**

id
titulo
descricao
image
prioridade
status
latitude
longitude
datahora

Figura 12: Tabela colaborações.

A tabela colaborações possui como chave primária o atributo numérico “id”. Os atributos “título”, “descrição”, “image” e “status” são do tipo texto. Os atributos “prioridade”, “latitude”, “longitude”, “datahora” são numéricos.

Os três serviços básicos desenvolvidos para a aplicação são: “Consultar colaborações”, “Criar colaboração” e “Atualizar colaboração”. Esses serviços consultam, inserem e alteram os dados da tabela colaborações. Exemplos das requisições feitas pelo aplicativo (no formato *json*) e das respostas dos serviços (também no formato *json*) são encontrados no Anexo C.

### 3.3.1 Consultar colaborações

O serviço responsável por consultar as colaborações recebe via *post* dados no formato *json*. Esses dados são as configurações atuais do aplicativo. As configurações são usadas para montar uma consulta que é feita no banco de dados. O resultado da consulta é devolvido para a aplicação também no formato *json*. Esse fluxo de execução é demonstrado no diagrama de atividades da Figura 13.

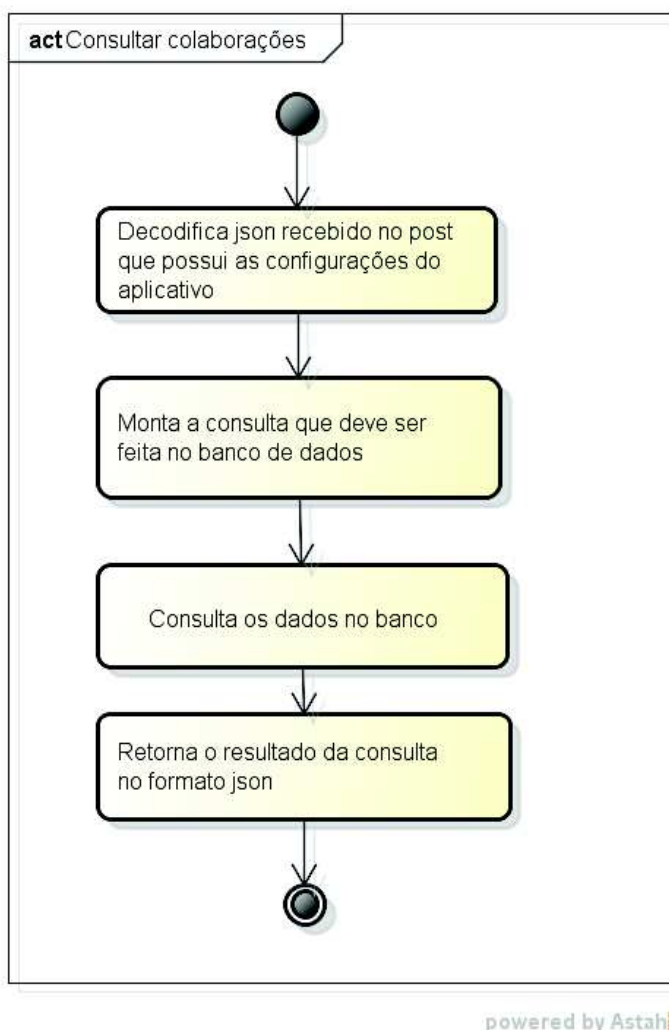


Figura 13: Diagrama atividades do serviço consultar colaborações.

### 3.3.2 Criar colaboração

O serviço responsável por criar uma colaboração recebe via *post* um *json* com as informações sobre a colaboração. Todas as informações recebidas são inseridas no banco de dados na tabela colaborações.

Opcionalmente o serviço pode receber uma foto (codificada em base 64) do local no qual a colaboração foi adicionada. Nesse caso, a foto é decodificada e salva em um diretório específico do servidor. Para que o nome do arquivo seja único, ele é sempre formado pelo prefixo “image” e o sufixo é o “id” da nova colaboração. Após a inserção da colaboração no banco de dados, caso a colaboração tenha uma foto, ela é atualizada com o nome do arquivo que foi salvo no servidor. Esse fluxo de execução é demonstrado no diagrama de atividades da Figura 14.

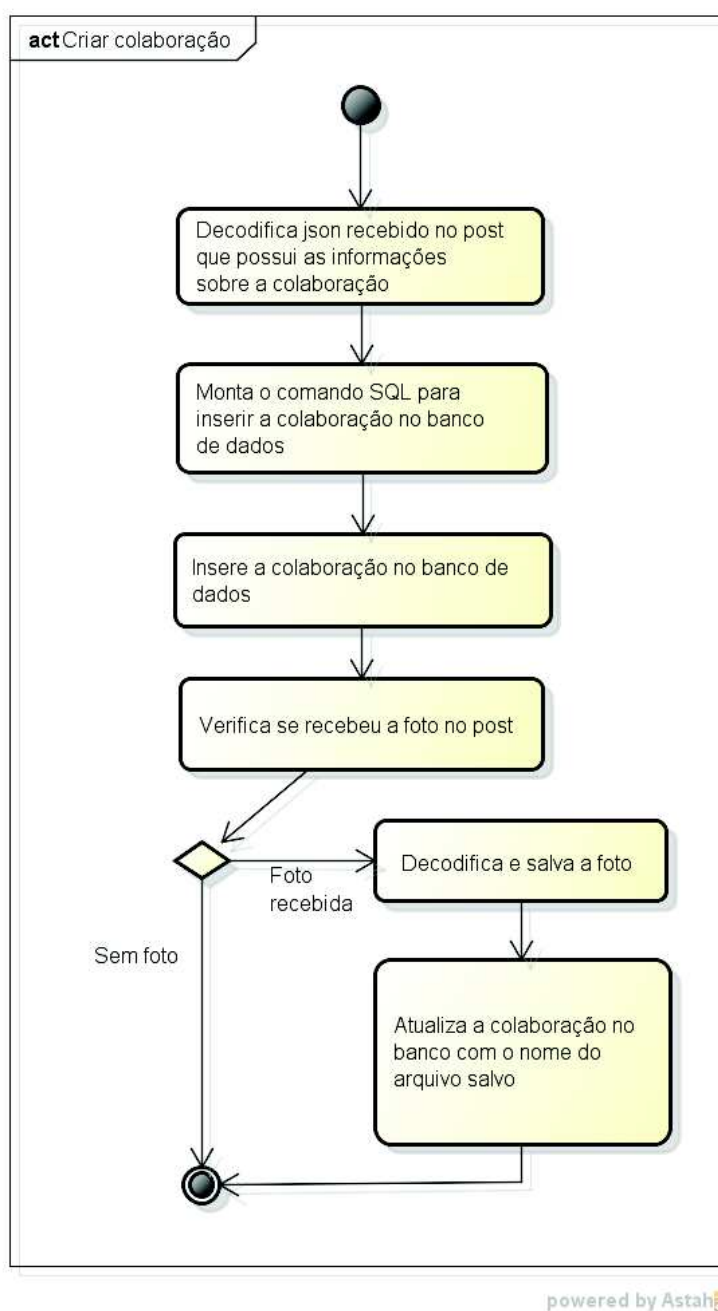


Figura 14: Diagrama atividades do serviço criar colaboração.

### 3.3.3 Atualizar colaboração

O serviço responsável por atualizar uma colaboração recebe via *post* um *json* com o identificador da colaboração e suas novas informações de prioridade e situação. As informações são atualizadas no banco de dados. Esse fluxo de execução é demonstrado no diagrama de atividades da Figura 15.

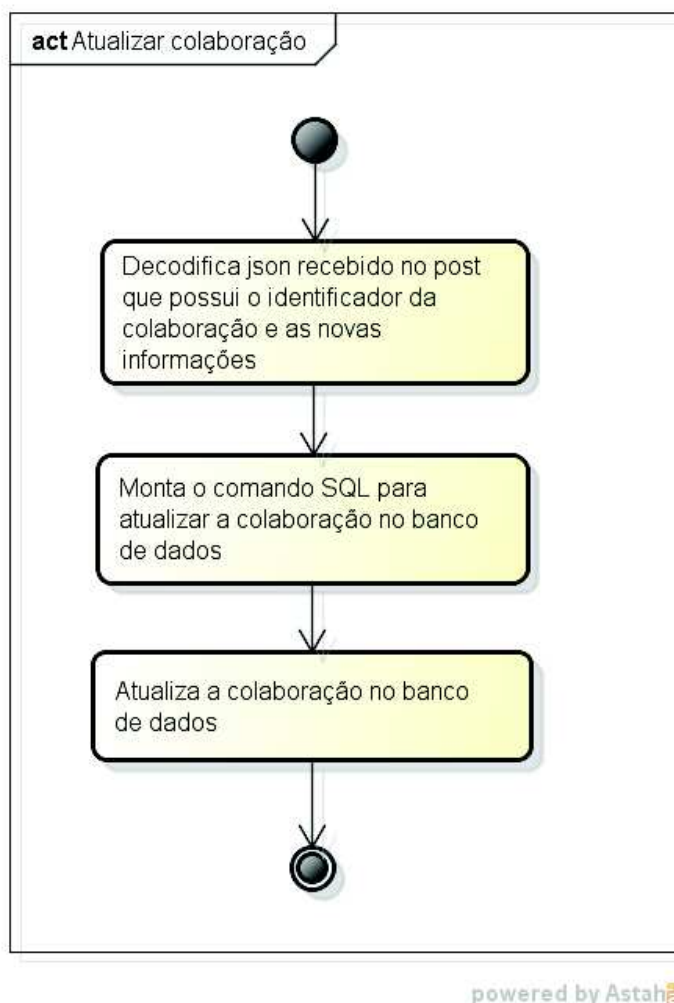


Figura 15: Diagrama atividades do serviço atualizar colaboração.

## 4 POSSIBILIDADES DE MELHORIAS DO APLICATIVO

Para futuras versões é interessante considerar a possibilidade de criar um cadastro de usuários e permissões. Dessa forma somente usuários com permissões específicas podem alterar a situação das colaborações. Usuários comuns podem continuar contribuindo de forma anônima, porém com a possibilidade de efetuar um



cadastro. Fazendo um cadastro, os usuários comuns podem ser avaliados e até mesmo receber recompensas por contribuições.

O usuário cadastrado recebe recompensas quando atingir um determinado número de colaborações verdadeiras. Ele também pode receber recompensas no caso de colaborações postadas que fizeram grande diferença para a sociedade. Essas recompensas poderiam ser feitas em forma de reconhecimento formal dentro do sistema ou até mesmo em retorno financeiro. Usuários com permissões especiais são responsáveis pelo gerenciamento de recompensas para os usuários comuns.

O aplicativo foi desenvolvido de forma a ser facilmente adaptado a outros assuntos não relacionados ao trânsito. Ele facilmente se adapta a outros tipos de serviços que exigem fiscalização de forma colaborativa. Mantem-se a arquitetura e apenas alteram-se as telas para o objetivo específico de fiscalização que é proposto pelo novo aplicativo.

## **5 CONCLUSÃO**

Entre os vários desafios na construção de um sistema colaborativo, pode-se concluir que o principal é a criação de um aplicativo no qual as pessoas se sintam motivadas em contribuir. Para o sistema proposto foram desenvolvidas apenas as funcionalidades básicas, portanto, o sistema precisaria ser incrementado para ser realmente usado. Com o objetivo de divulgar e criar o interesse e motivação das pessoas no uso do aplicativo, o sistema poderia, por exemplo, ser incrementado com a integração com alguma rede social.

### **CROWDSOURCING SYSTEM TO HELP TRAFFIC OFFICERS**

**Abstract:** This paper describes the construction of a crowdsourcing system to help traffic officers. The system's goal is to help the laws to be more respected and give some orientation to the traffic officers about the key places of the city where there are more traffic jams. Anyone can help with information about infractions that are happening, the places where they happen the most, or just traffic issues due, for example: traffic jams, broken semaphores, traffic accidents, etc. The collaborations are saved in the system and they are available to be queried by the traffic officers. This way, traffic officers can see in real-time the issues and infractions that are happening in the city and take an action faster and smarter. The citizen contributes with important information for the general good of society, helping in the maintenance of the traffic and increasing the sense of citizenship. The paper

details the challenges encountered in the construction of a crowdsourcing system, the solutions adopted and the results achieved.

Keywords: Crowdsourcing. Mobile Devices. Android.

## REFERÊNCIAS

BRABHAM, Daren. **Crowdsourcing as a Model for Problem Solving**. 2008.

BUTTON, Kenneth; HENSHER, David. **Handbook of Transport Systems and Traffic Control**. Elsevier Science, 2001.

INFO. **Apps para ajudar no trânsito**. Disponível em:  
<<http://info.abril.com.br/noticias/downloads/fotonoticias/8-apps-para-ajudar-no-transito.shtml>>. Acesso em: 11 jul. 2013.

ANDROID DEVELOPER GUIDE. **What is API Level?**, Disponível em:  
<<http://developer.android.com/guide/topics/manifest/uses-sdk-element.html>>.  
Acesso em: 15 ago. 2013.

WIKIPEDIA. **Haversine formula**. Disponível em:  
<[http://en.wikipedia.org/wiki/Haversine\\_formula](http://en.wikipedia.org/wiki/Haversine_formula)>. Acesso em: 1 set. 2013.

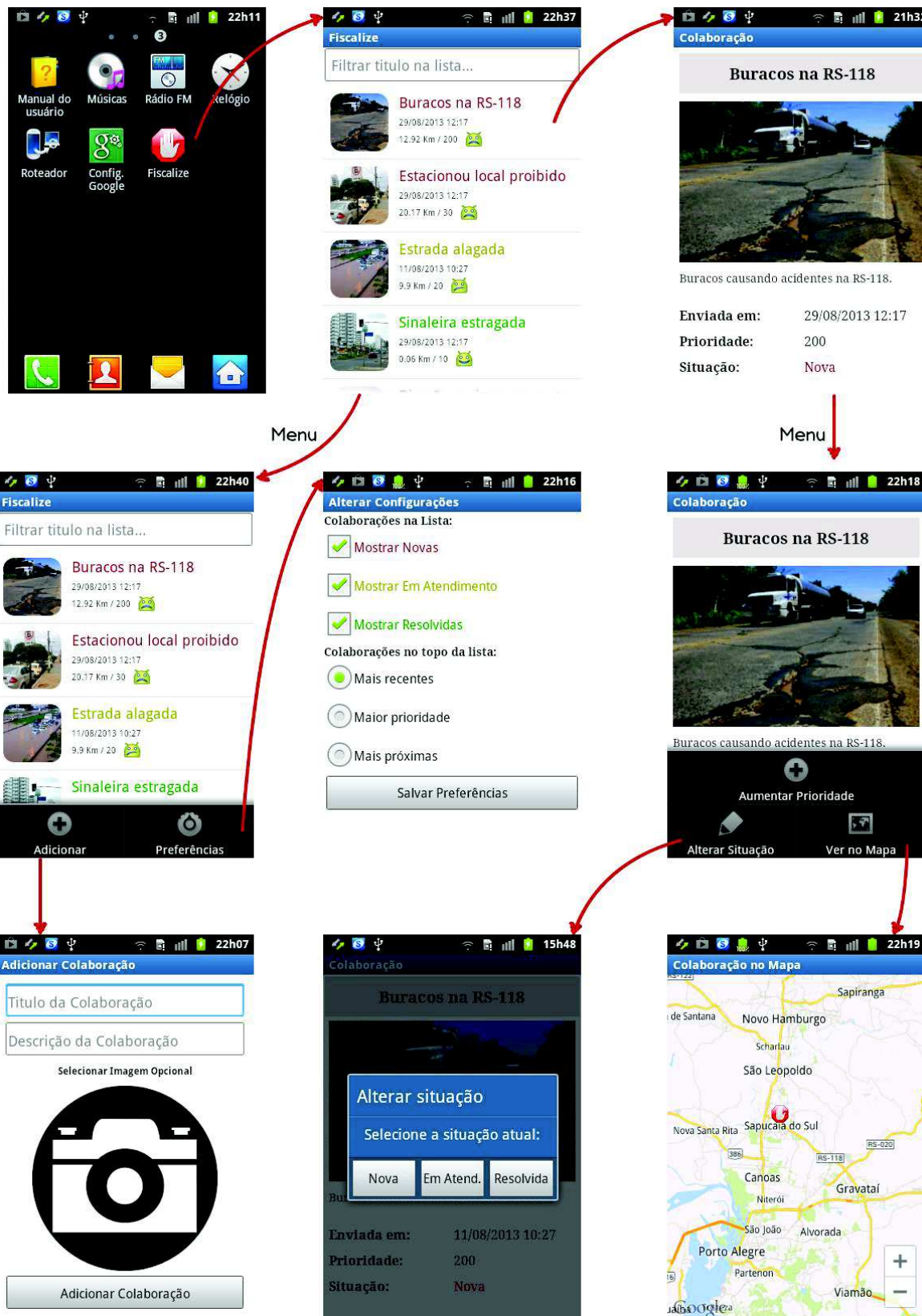
UIL. **Universal Image Loader for Android**. Disponível em:  
<<https://github.com/nostra13/Android-Universal-Image-Loader>>. Acesso em: 1 set. 2013.

ANDROID DEVELOPER GUIDE. **Using Shared Preferences**. Disponível em:  
<<http://developer.android.com/guide/topics/data/data-storage.html>>. Acesso em: 15 ago. 2013.

ANDROID DEVELOPER GUIDE. **Google Maps API V2**. Disponível em:  
<<https://developers.google.com/maps/documentation/android/>>. Acesso em: 15 ago. 2013.

## ANEXOS

### ANEXO A – Principais fluxos de navegação entre telas do aplicativo





ANEXO C – Exemplos de requisições e respostas *json*

Exemplo 1: Consultar colaborações.

Requisição:

```
{
  "max_colaboracoes": 20,
  "resolvidas": true,
  "novas": true,
  "em_atendimento": true,
  "latitude": -29.937969,
  "longitude": -51.075988,
  "order_by": "datahora"
}
```

Resposta:

```
{
  "colaboracoes": [
    {
      "id": "1",
      "titulo": "Titulo 1",
      "descricao": "Descricao 1",
      "image": "image1.jpg",
      "prioridade": "2222",
      "status": "Nova",
      "latitude": "53.551",
      "longitude": "9.993",
      "datahora": "1336226501635",
      "distance": "48.19900135561273"
    },
    {
      "id": "2",
      "titulo": "Titulo 2",
      "descricao": "Descricao 2",
      "image": "image2.jpg",
      "prioridade": "111",
      "status": "Resolvida",
      "latitude": "53.558",
      "longitude": "9.927",
      "datahora": "1336226501635",
      "distance": "48.19900135561273"
    }
  ]
}
```

Exemplo 2: Atualizar colaboração.

```
{
  "colaboracao": {
    "id": "1",
    "status": "Em atendimento",
    "prioridade": "2223"
  }
}
```



Exemplo 3: Criar colaboração.

```
{
  "colaboracao": {
    "image": "",
    "longitude": -51.1554571,
    "titulo": "Titulo 1",
    "latitude": -29.7930237,
    "descricao": "Descricao 1",
    "datahora": 1376443613090
  }
}
```