

# Finder: Aplicativo de Achados e Perdidos

Justh Franklin M. Leal<sup>1</sup>, Gustavo Bartz Guedes<sup>1</sup>

<sup>1</sup>Instituto Federal de São Paulo (IFSP) Campus Hortolândia  
Caixa Postal 15.064 – 91.501-970 – Hortolândia – SP – Brasil

justhfranklin@hotmail.com, gubartz@ifsp.edu.br

**Abstract.** *The loss of an object or something of value can cause prejudice. Many times those who find an object have difficulties to encounter the owner. Likewise, those who lost something also feel the need to report the loss in the expectation of finding the object. The Finder application allows the register of lost and found objects and shares the information to facilitate their return. In addition, Finder uses the Global Positioning System (GPS) to indicate where an object was lost or found.*

**Resumo.** *A perda de um objeto ou algo de valor pode causar grande prejuízo e diversos transtornos. Muitas vezes quem encontra um objeto tem dificuldade em devolvê-lo ao dono. Da mesma maneira, quem perdeu algo também sente a necessidade de informar a perda na expectativa de encontrar o objeto. O aplicativo Finder permite cadastrar objetos perdidos e encontrados e compartilhar as informações para facilitar o retorno de itens perdidos. O Finder utiliza o Global Positioning System (GPS) para indicar o local em que um objeto foi perdido ou encontrado.*

## 1. Introdução

Perder algo pode resultar em consequências financeiras e emocionais causando diversos transtornos na vida de quem passa por isso. Os Correios disponibilizam à população o serviço “Achados e Perdidos”, que funciona em todo o Brasil, mas é concentrado nas agências centrais das capitais. Existe uma procura grande pelo serviço, mas somente 5% dos itens, sob a guarda dos Correios, são resgatados, uma vez que é preciso se dirigir até uma agência [Nantes 2012]. Com o intuito de ajudar pessoas que têm seus itens perdidos a localizá-los, foi desenvolvido o aplicativo *Finder* que mantém o cadastro de itens que foram perdidos ou encontrados e permite marcar a localização dessas ocorrências.

A utilização de dispositivos móveis tem crescido devido ao aumento do poder computacional dos aparelhos e pela praticidade em se realizar tarefas cotidianas. Segundo [Russell 2017] as pessoas estão consumindo mais conteúdos pelos *smartphones* do que em computadores, que estão cada vez mais destinados a áreas de trabalho do que uso doméstico. O uso de dispositivos móveis para auxiliar na busca por itens pode facilitar que objetos ou documentos perdidos sejam encontrados pelo compartilhamento de informações.

## 2. Metodologia

Inicialmente foi criado um protótipo do aplicativo *Finder*, a partir disso, foi evoluído para aperfeiçoá-lo com novas funcionalidades. Os requisitos foram extraídos por meio de pesquisa em aplicativos correlatos de busca por itens perdidos.

O *Feature Driven Development* (FDD) foi a metodologia ágil utilizada no desenvolvimento do *Finder* sendo focada na entrega por funcionalidades. O desenvolvimento do aplicativo foi dividido em três partes principais. Cada parte foi subdividida em funcionalidades passíveis de entrega em no máximo duas semanas.

A lista a seguir apresenta as três partes principais no desenvolvimento do *Finder*.

1. Elaboração do esquema da Base de dados;
2. Especificação e desenvolvimento do *web service*;
3. Projeto das telas do aplicativo.

Para minimizar a incidência de erros foram realizados testes de utilização ao final do desenvolvimento de cada tela. O código era corrigido e a funcionalidade retestada até que estivesse em conformidade com o teste inicialmente especificado.

### 3. Aplicativos correlatos

Foi realizada uma busca, no site da Loja de Aplicativos *Play Store* [Google 2017e], por aplicativos de busca por itens perdidos. Foram encontrados dois aplicativos: “Perdi, mas achei” e o “Objetos perdidos y encontrados”. Porém, somente o “Perdi, mas achei” está no idioma português do Brasil.

O aplicativo “Perdi, mas achei” foi a primeira plataforma digital de achados e perdidos do Brasil [Google 2017d]. Ele disponibiliza o cadastro dos itens perdidos e encontrados e a listagem para todos os usuários cadastrados. Entretanto, foram encontradas algumas falhas no funcionamento do aplicativo e alguns itens que podem ser melhorados conforme mostra a listagem a seguir.

1. Falha ao tentar realizar o login após a operação de *logout*;
2. Falhas de apresentação dos itens em tela quando selecionados;
3. Ícones muito pequenos comprometendo a usabilidade;
4. Listagem dos itens sem ordenação e sem a possibilidade de filtrar o resultado.

O aplicativo “Objetos perdidos y encontrados” está disponível no idioma espanhol e funciona com o cadastro de itens perdidos e encontrados e permite, com a utilização do GPS, marcar a localização do item. As marcações podem ser visualizadas em um mapa que apresenta todas as ocorrências cadastradas. Este aplicativo foi o mais completo em termos de recursos identificado na pesquisa.

A Figura 1 mostra uma comparação das funcionalidades entre os aplicativos encontrados.

		
➤ Login e registro de usuários	✓	✓
➤ Cadastro de itens encontrados e perdidos	✓	✓
➤ Filtro de itens cadastrados por usuário	✗	✗
➤ Listagem de itens cadastrados	✓	✗
➤ Exclusão de itens cadastrados	✓	✓
➤ Utilização de Geolocalização para cadastro e acesso local	✗	✓
➤ Categorização de itens	✓	✓
➤ Notificação de novo item cadastrado	✓	✓
➤ Visualização características item	✓	✓

**Figura 1. Comparativo funcionalidades Perdi, mas Achei e Objetos perdidos y encontrados.**

Dentre as funcionalidades do “Perdi, mas achei” só não foram incorporadas ao *Finder* a “Categorização dos itens” que é uma maneira de classificar o item no momento do cadastro e “Notificação de novo item cadastrado”. Esses recursos podem ser inseridos como melhoria futura ao aplicativo. Em contrapartida, foi adicionada ao *Finder* à utilização do *Google Maps* [Google 2017b] para marcar a geolocalização do item da ocorrência e o “Filtro de itens cadastrados por usuário” que facilita a busca por itens.

#### 4. Referencial Teórico

Esta seção apresenta o referencial teórico dos conceitos que foram estudados para o desenvolvimento do *Finder*.

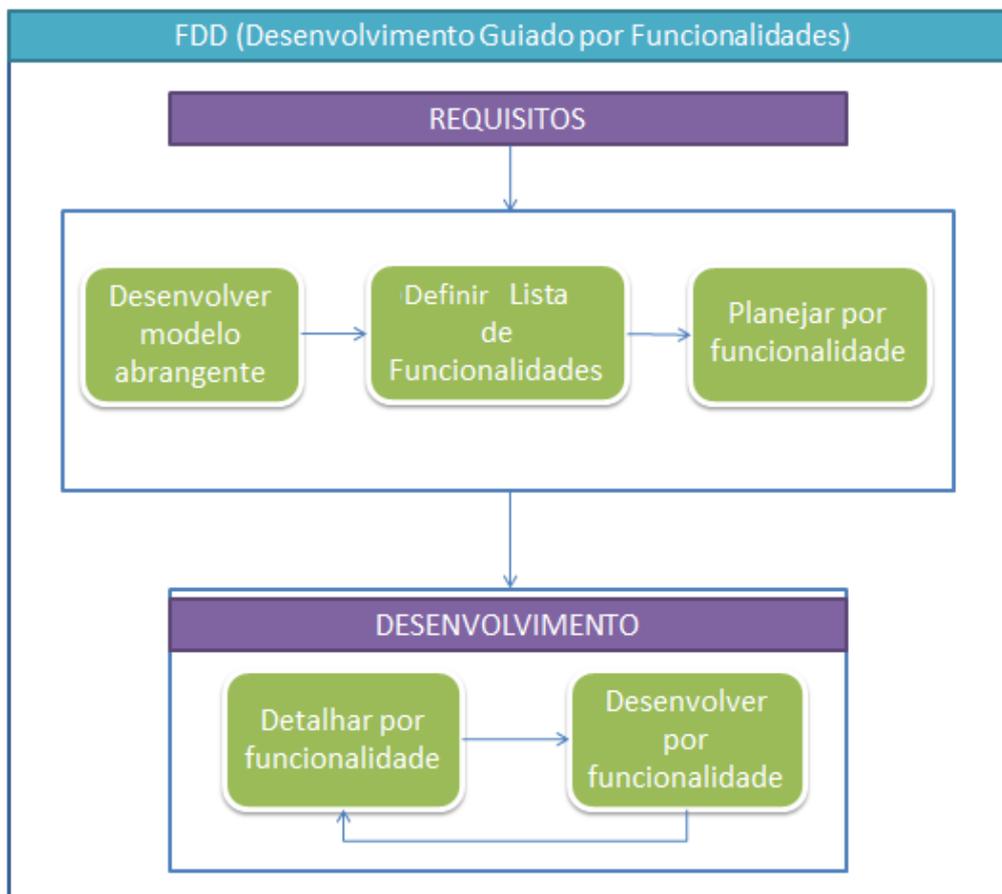
##### 4.1. *Feature Driven Development* (FDD)

Desenvolvimento Guiado por Funcionalidades (FDD) é um método ágil apoiado na elaboração do aplicativo com base em uma lista de requisitos funcionais principais. Cada funcionalidade é chamada de *feature*, que após identificada é utilizada para planejar como será o processo de desenvolvimento de acordo com o nível de prioridade de cada uma [Martins 2007].

O desenvolvimento no FDD baseia-se em cinco processos sequenciais apresentados na Figura 2.

1. Desenvolver o Modelo Abrangente: elaborar um modelo conceitual do sistema e objetivos.
2. Definir lista de funcionalidades: definir uma lista detalhada e ordenada hierarquicamente das funcionalidades que definem cada etapa de desenvolvimento.

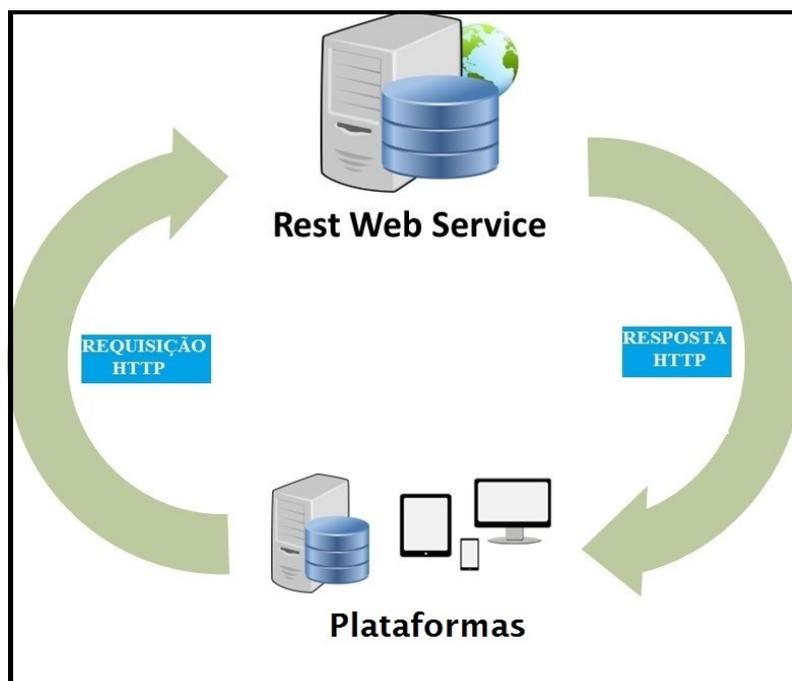
3. Planejar por funcionalidade: definir os recursos, padrões e regras atribuídas a cada funcionalidade.
4. Detalhar por funcionalidade: definir e discriminar ações da funcionalidade na sequência de implementação.
5. Desenvolver por Funcionalidade: implementar os requisitos de acordo com a sequência definida.



**Figura 2. Fluxo de Atividades FDD.**

#### **4.2. Representational State Transfer (REST)**

REST é a sigla para “Transferência de estado representacional”, uma arquitetura de *software* sobre um serviço operado em rede com o uso de *web services*. Cada serviço é fornecido para consumo de modo independente para que distintas plataforma realizem o acesso e consigam utilizar os recursos por meio do protocolo *Hypertext Transfer Protocol* (HTTP). As requisições recuperam ou alteram o estado (conteúdo) do recurso disponibilizado [Fielding 2000]. A Figura 3 ilustra a implementação da Arquitetura REST.



**Figura 3. Fluxo - Arquitetura REST**

É possível realizar diversas ações sobre os recursos utilizando os métodos disponíveis pelo protocolo HTTP.

1. POST: Cria conteúdo do recurso.
2. GET: Recupera o conteúdo do recurso.
3. DELETE: Remove o conteúdo do recurso.
4. PUT: Atualiza o conteúdo do recurso.

REST é indicado para aplicações que possuem limitações de recursos e de largura de banda, isso caracteriza uma ótima opção para utilização em dispositivos móveis que podem sofrer com instabilidades de conexão de internet.

### **4.3. Geolocalização**

Geolocalização é o posicionamento que define a localização de algo em um sistema determinado de coordenadas e pode ser obtida por dispositivos móveis ou plataformas capazes de fazer referência por latitude e longitude representadas no globo terrestre.

Segundo [Bohrer 2011], nos dispositivos móveis, a localização pode ser encontrada pelo Sistema de Posicionamento Global (GPS) com margem de erro de poucos metros. Porém, seu funcionamento pleno depende de um ambiente ao ar livre e pode haver demora no retorno das informações de localização devido a necessidade de sincronia com o satélite. Outra forma de obter a geolocalização é utilizando a internet com a triangulação das torres de telefonia celular e redes sem fio (*Wi-fi*). A Figura 4 apresenta o funcionamento do GPS utilizando celular, em que a conexão pode ser estabelecida utilizando torres de transmissão de internet, satélite ou uma combinação dos dois garantindo maior eficiência no funcionamento.



Figura 4. Funcionamento GPS [AndroidZ 2013].

#### 4.4. Google Maps Android API

O Google Maps é um serviço fornecido pelo Google e permite que um usuário utilize mapas da maior parte do mundo pelos dispositivos móveis e computadores. Ele fornece uma *Application Programming Interface* (API) para a integração dos mapas com a aplicação.

Utilizando a Google Maps Android API é possível incluir mapas diretamente no aplicativo. A API processa automaticamente o acesso aos servidores e realiza a sincronização dos dados de acordo com as ações realizadas. Também é possível adicionar marcadores, polígonos e sobreposições a um mapa e permite ao usuário alterar a visualização de determinada área [Google 2017c].

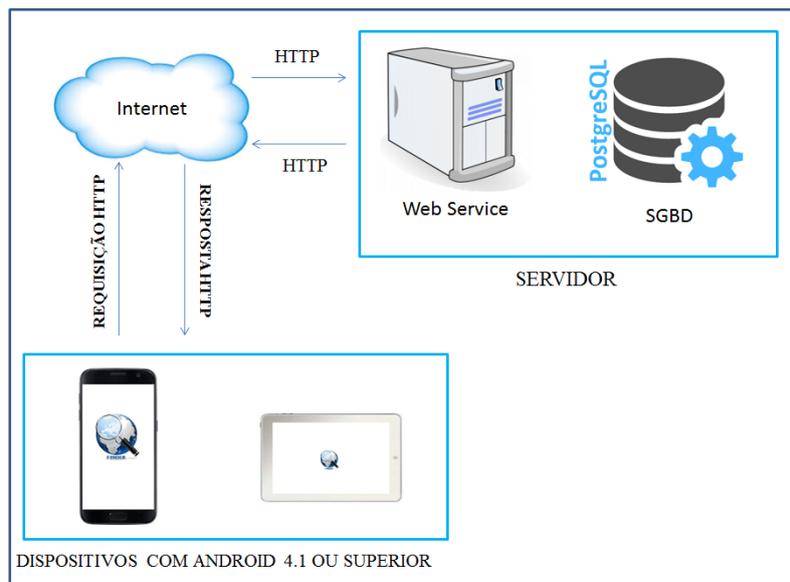
### 5. Desenvolvimento do Finder

Nesta seção é apresentado o desenvolvimento, telas e funcionamento do aplicativo *Finder*. Foi desenvolvido um protótipo das telas como modelo conceitual para elaboração do aplicativo e a partir disso foi planejado a implementação dos requisitos.

#### 5.1. Arquitetura

O aplicativo foi desenvolvido utilizando a arquitetura de *Web Services*, implementados para utilização em dispositivos móveis por meio da rede internet. Por meio da arquitetura REST as informações podem ser transmitidas entre os dispositivos que utilizem o *Finder* por meio do protocolo HTTP. A arquitetura é apresentada na Figura 5 sendo composta pelos itens listados a seguir.

1. Dispositivos Android: qualquer dispositivo com sistema operacional Android 4.1 ou superior.
2. Servidor: provê a camada de persistência e de serviços para o aplicativo. Respetivamente, utiliza o Sistema de Gerenciamento de Banco de Dados PostgreSQL e a linguagem de programação PHP.



**Figura 5. Arquitetura Finder.**

A composição das funcionalidades do *Finder* foi elaborada compondo todos recursos de acordo a Figura 6.

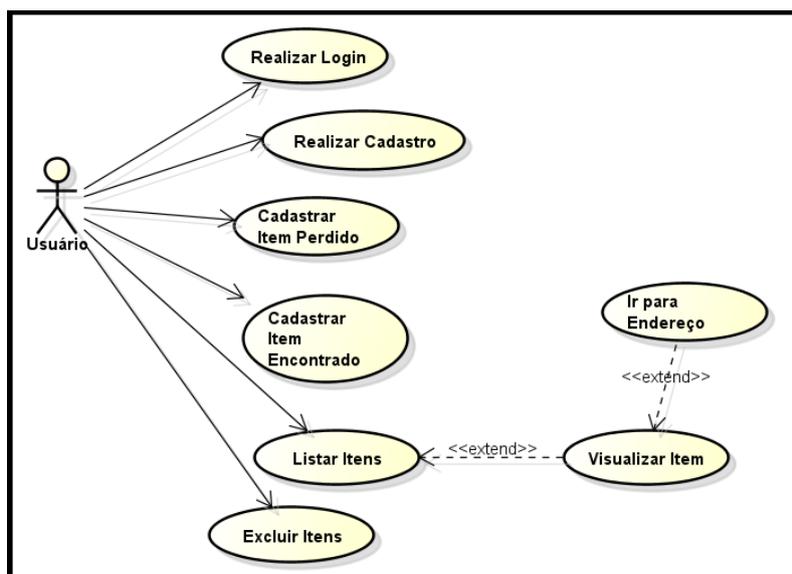
<b>FDD</b>	
<b>FEATURES DESENVOLVIMENTO FINDER</b>	
Registrar usuário	
Validar Usuário Duplicado	
Realizar Login	
Validar Login	
Cadastrar Item Encontrado	
Cadastrar Item Perdido	
Salvar Foto e Imagem	
Listar Itens	
Listar Itens do Usuário	
Excluir Item	
Gravar Localização	
Criar Rota GPS	

**Figura 6. FDD Finder.**

O Diagrama de Caso de Uso, apresentado na Figura 7, foi elaborado de acordo com os requisitos e funcionalidades apresentadas a seguir.

1. Realizar Cadastro: o usuário realiza seu cadastro informando seus dados e credenciais de login.
2. Realizar Login: o usuário realizar sua autenticação por meio de e-mail e senha cadastradas.
3. Cadastrar Item Perdido: cadastro do item perdido informando a descrição, possível localização e foto.
4. Cadastrar Item Encontrado: cadastro de um item perdido e que foi encontrado. A descrição, localização e uma foto do item são informados.

5. Listar Itens: apresenta a listagem de todos os itens cadastrados podendo ser filtrados por “Perdidos” ou “Encontrados”.
6. Excluir itens: o usuário pode excluir um item que tenha cadastrado.
7. Visualizar Item: apresenta dados detalhados, quando um item é selecionado, como o contato de quem o cadastrou.
8. Ir para endereço: Quando o item estiver a mostra e este contiver a localização cadastrada o aplicativo aciona o (google maps) do dispositivo com o trajeto até o endereço registrado.



**Figura 7. Diagrama de Caso de Uso.**

## 5.2. Tecnologias utilizadas

A Figura 8 resume as tecnologias utilizadas no processo de desenvolvimento do aplicativo.

Nome	Descrição	Versão
Android Studio	IDE de desenvolvimento de Aplicativos Android	2.2.2
PostgreSQL	Sistema de Gerenciamento de Banco de Dados	9.5.6
PHP	Linguagem utilizada para elaboração do servidor	5.5.30
Slim Framwork	Microframework de auxílio ao PHP para gerenciamento das rotas do web service	2.6.3

**Figura 8. Tecnologias utilizadas no *Finder*.**

O Android Studio é a ferramenta oficial para desenvolvimento de aplicativos para Android. Dentre os recursos fornecidos o emulador *Android Virtual Device* (AVD) permite simular um *smartphone* em uma máquina virtual [Google 2017a]. Este recurso foi utilizado para testar a implementação dos requisitos a cada iteração no desenvolvimento do projeto.

A implementação da camada de persistência dos dados do *web service* foi realizada no Sistema de Gerenciamento de Bancos de Dados Objeto-Relacional (SGBDOR)

PostgreSQL. Sua escolha se deve ao fato de possuir licença de uso gratuito e a gama de recursos disponíveis, além de contar com uma arquitetura comprovada por anos de desenvolvimento que proporcionou ao PostgreSQL confiabilidade integridade de dados e correção [Group 2017].

O *Web Service* foi desenvolvido na linguagem PHP [PHP 2017] em conjunto com o *microframework* Slim que implementa um sistema de rotas utilizando o padrão *Front-Controller*. Com isso, as solicitações e respostas HTTP são tratadas de maneira centralizada e organizada [Team 2017].

### 5.3. Interface

A interface foi desenvolvida com o uso de *Fragments* em que cada tela é definida como um fragmento independente, sendo necessário manter apenas uma *Activity* principal [Google 2016]. A Figura 9 mostra os fragmentos implementados.

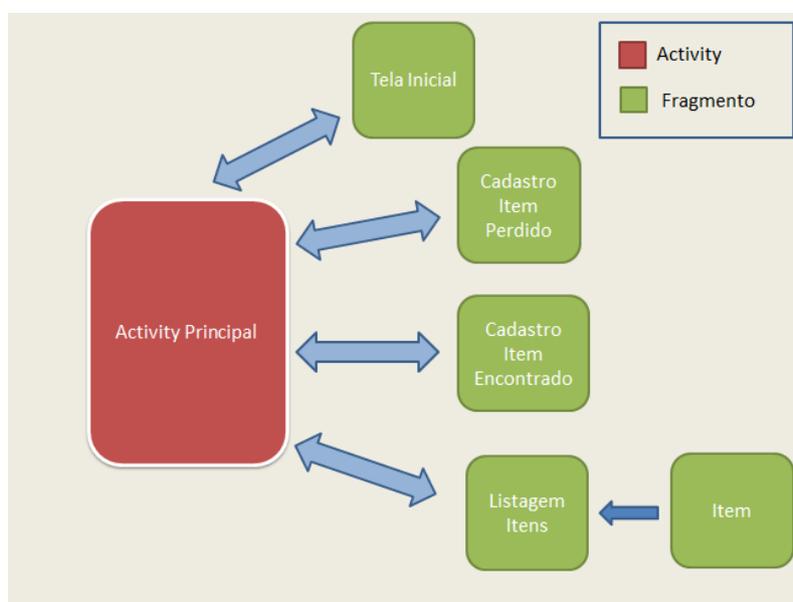
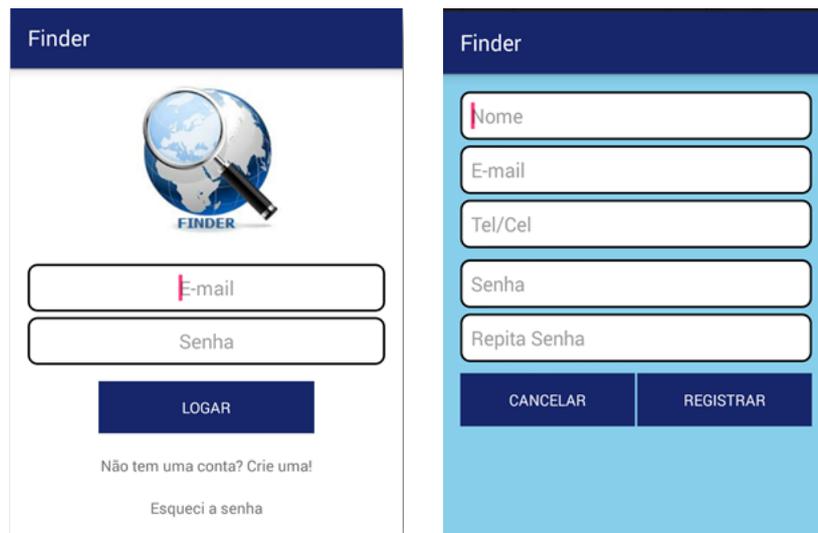


Figura 9. Composição de fragmentos no Finder.

Quando o usuário aciona uma ação referente a um Fragmento, este é inserido na *Activity* principal. Essa operação é denominada de “inflar um fragmento”. Com isso, partes da tela podem ser substituídas por outros fragmentos. Para navegar pelos itens existem dois fragmentos. Um para a listagem dos itens e outro para mostrar seus detalhes, que é acionado quando um item é selecionado pelo usuário.

### 5.4. Login e Registro

A Figura 10 apresenta a tela de login e de cadastro de usuário. A autenticação é realizada com o “E-mail” e “Senha” previamente cadastrados, sendo possível recuperar a senha em caso de perda. Também é possível acessar a tela de cadastro pela opção “Não tem uma conta? Crie uma!”.



**Figura 10. Tela de Login e de Registro.**

### 5.5. Cadastro de Itens

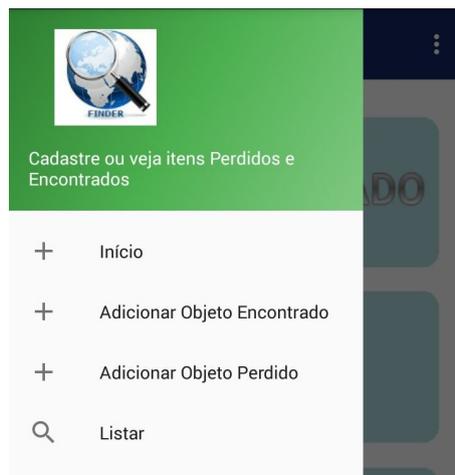
A Figura 11 apresenta o menu principal do *Finder* com os acessos às suas funcionalidades, descritas a seguir.

1. Encontrado: abre a listagem de itens encontrados cadastrados.
2. Perdido: abre a listagem de itens perdidos cadastrados.
3. Listar Itens: abre a tela com a listagem de todos os itens cadastrados, independentemente de serem itens perdidos ou encontrados.



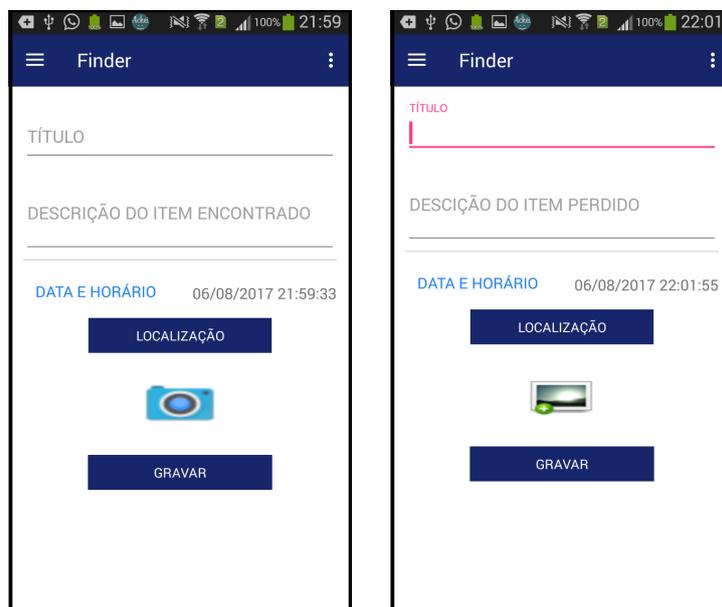
**Figura 11. Tela Principal Finder.**

As opções também estão disponíveis no menu lateral (*Navigation Drawer Menu*) conforme mostra a Figura 12.



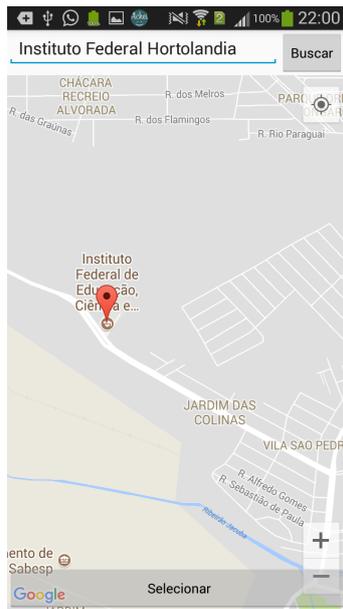
**Figura 12. Navigation Drawer Menu Finder.**

A Figura 13 apresenta as telas de cadastro de item encontrado e de item perdido. As duas telas possuem os mesmos campos, o que as diferencia é a inserção de foto. No cadastro de item encontrado o aplicativo de foto é aberto para que uma foto possa ser tirada. Já no cadastro de item perdido a galeria de imagens é aberta para a seleção da foto do objeto perdido, caso exista.



**Figura 13. Telas de cadastro de item perdido e item encontrado Finder.**

Nas duas telas de cadastro dos itens existe a possibilidade de marcar a localização por busca de endereço ou utilizando a localização atual conforme ilustra a Figura 14.



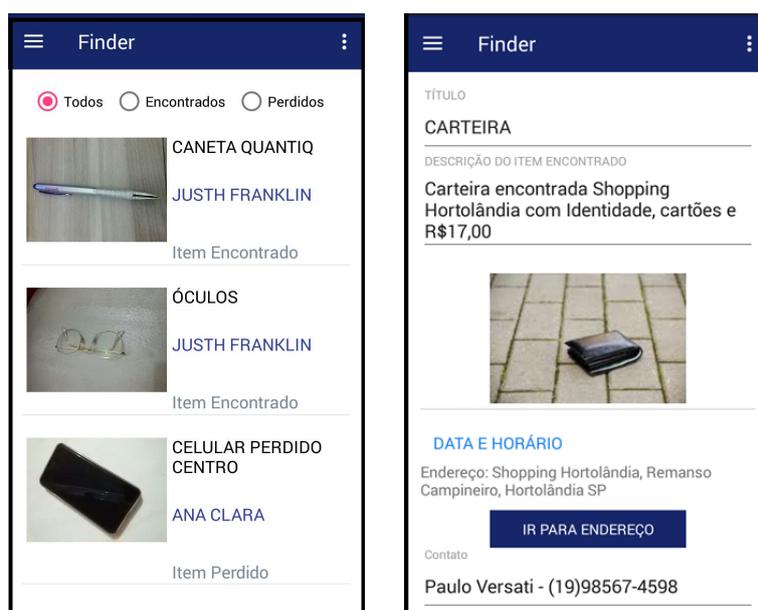
**Figura 14. Marcação de localização.**

É possível excluir um item cadastrado pelo menu lateral na opção “Meus Itens”, o *Finder* apresentará todos os itens do usuário. Após a exclusão o item é removido do banco de dados e não é mais possível revê-lo.

## **5.6. Listagem e visualização de Itens**

A listagem dos itens cadastrados apresenta todos os registros com ordenação decrescente por data de cadastro. Para cada item é mostrado um título, o usuário que realizou o cadastro e um indicativo de objetivo “Perdido” ou “Encontrado”. Na Figura 15, a imagem da esquerda apresenta a tela de listagem. Já a da direita mostra a tela de detalhes, quando um usuário seleciona um item específico.

Uma vez que o item for selecionado para visualização é possível traçar uma rota utilizando o botão “Ir para o endereço”. Esse ativa o GPS do dispositivo, aciona o *google maps* e apresenta a rota com base no endereço cadastrado.



**Figura 15. Tela de listagem de itens (esquerda) e Tela de visualização do item (direita).**

## 6. Conclusão

O *Finder* foi finalizado e é um aplicativo capaz de auxiliar na busca por itens perdidos por disponibilizar uma plataforma em que pessoas compartilham informações com foto e descrição do que foi perdido ou encontrado. Além disso, a utilização do GPS com o uso do *Google Maps* determina o local em que um item foi perdido ou encontrado, com a possibilidade de traçar um rota para esses locais.

A utilização do *Slim Framework* na implementação dos *Web Services* disponibiliza uma padronização de implementação das rotas para os *web services* garantindo uma melhor manutenibilidade e organização do código.

Uma dificuldade encontrada no desenvolvimento ocorreu na busca por documentação referente ao desenvolvimento para Android. Apesar de existirem muitas fontes as versões de API são muito distintas, fazendo com que muitos exemplos tenham comportamentos diferentes no momento dos testes ou até que não funcionem por completo.

Como possíveis implementações futuras o *Finder* pode ser complementado com recursos de notificações automáticas de objetos perdidos e encontrados por regiões; criar categoria para os itens, pois pode melhorar o uso de filtros na listagem dos itens. Outra possibilidade é a conciliação automatizada entre o cadastro de itens encontrados e perdidos para que apontar que descrições semelhantes se refiram aos mesmos itens. Por fim, é possível determinar possíveis localizações dos itens pela análise das coordenadas, tudo isso pode contribuir para tornar o *Finder* mais completo e auxiliar ainda mais os usuários.

## Referências

AndroidZ (2013). Diferença entre gps e a-gps e como funciona? <http://www.androidz.com.br/portal/diferenca-entre-gps-e-a-gps-e-como-funciona.html>. Recuperado em 08 de outubro, 2017].

- Bohrer, F. J. (2011). Serviço de geolocalização para plataforma android. Master's thesis, UNIVATES, Lajeado.
- Fielding, R. T. (2000). Estilos arquitetônicos e o design de arquiteturas de software baseadas em rede. Irvine.
- Google (2016). Fragmentos. <https://developer.android.com/guide/components/fragments.html?hl=pt-br>. Recuperado em 08 de outubro, 2017].
- Google (2017a). Create and manage virtual devices. <https://developer.android.com/studio/run/managing-avds.html?hl=pt-br>. Recuperado em 08 de outubro, 2017].
- Google (2017b). Google maps. <https://www.google.com.br/maps>. Recuperado em 4 de setembro, 2017].
- Google (2017c). Introdução à google maps android api. <https://developers.google.com/maps/documentation/android-api/intro?hl=pt-br>. Recuperado em 08 de outubro, 2017.
- Google (2017d). Perdi mas achei. <https://play.google.com/store/apps/details?id=com.perdimasachei&hl=pt-br>. Recuperado em 01 de outubro, 2017].
- Google (2017e). Play store. <https://play.google.com/store?hl=pt-br>. Recuperado em 01 de Outubro, 2017].
- Group, P. D. (2017). About postgresql. <https://www.postgresql.org/about/>. Recuperado em 15 de agosto, 2017.
- Martins, J. C. C. (2007). *Técnicas para gerenciamento de projetos de software*. Brasport.
- Nantes, C. (2012). Cidadão pode acessar serviço de achados e perdidos dos correios pela internet. <http://www.brasil.gov.br/infraestrutura/2012/02/cidadao-pode-acessar-servico-de-achados-e-perdidos-dos-correios-pela-internet>. Recuperado em 10 de outubro, 2017.
- PHP (2017). O que é o php? [https://secure.php.net/manual/pt\\_BR/intro-what-is.php](https://secure.php.net/manual/pt_BR/intro-what-is.php). Recuperado em 10 de setembro, 2017.
- Russell, J. (2017). Report: Android overtakes windows as the internet's most used operating system. [https://techcrunch.com/2017/04/03/statcounter-android-windows/?ncid=rss&utm\\_source=feedburner&utm\\_medium=feed&utm\\_campaign=Feed%3A+Techcrunch+%28TechCrunch%29](https://techcrunch.com/2017/04/03/statcounter-android-windows/?ncid=rss&utm_source=feedburner&utm_medium=feed&utm_campaign=Feed%3A+Techcrunch+%28TechCrunch%29). Recuperado em 01 de Outubro, 2017].
- Team, S. F. (2017). Slim. <https://www.slimframework.com/docs>. Recuperado em 15 de agosto, 2017.