UNIVERSIDADE FEDERAL DO ESPIRITO SANTO
CENTRO DE CIÊNCIAS AGRÁRIAS
DEPARTAMENTO DE COMPUTAÇÃO

FELIPE DE OLIVEIRA MIERTSCHINK

APLICATIVO MÓVEL PARA OTIMIZAÇÃO DE GASTOS DE COMBUSTÍVEIS

ALEGRE-ES
2015

FELIPE DE OLIVEIRA MIERTSCHINK

APLICATIVO MÓVEL PARA OTIMIZAÇÃO DE GASTOS DE COMBUSTÍVEIS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espirito Santo – UFES como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientador : Prof. M.Sc. Edmar Hell Kampke

ALEGRE-ES

2015

FELIPE DE OLIVEIRA MIERTSCHINK

APLICATIVO MÓVEL PARA OTIMIZAÇÃO DE GASTOS DE COMBUSTÍVEIS

Trabalho de conclusão de curso apresentado ao Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em 02 de dezembro de 2015.

COMISSÃO EXAMIDADORA

Prof. M.Sc. Edmar Hell Kampke
Universidade Federal do Espírito Santo
Orientador

Prof. M.Sc. Rodrigo Freitas Silva
Universidade Federal do Espírito Santo

Profª. M.Sc. Simone Dornelas Costa
Universidade Federal do Espírito Santo

AGRADECIMENTOS

A Deus, por ter me dado a oportunidade de chegar até aqui com saúde e por sempre me mostrar qual o melhor caminho a percorrer.

A meu pai Edgar Miertschink e a minha mãe Lucienne Caetano de Oliveira Miertschink por sempre me apoiarem em todos os momentos da minha vida, dando conselhos, incentivando, dando broncas e cuidando de mim.

A minha namorada e melhor amiga Bruna Machado Neves, por sempre estar ao meu lado nesses últimos 4 anos e meio, me aguentando, principalmente nos últimos 3 meses, me apoiando, trocando ideias e me incentivando.

A meu orientador Edmar Hell Kampke, pelo comprometimento e atenção nas orientações.

A todos meus amigos, por me proporcionarem momentos de grande alegria e me apoiarem em momentos de tristeza durante todo esse tempo de UFES.

Aos “DFs” por todas as jogatinas, churrascos e tempo passado junto comigo nesses anos de faculdade.

A meus irmãos de republica, que foram uma segunda família para mim.

A todos meus familiares, pelo carinho e apoio.

Aos professores Rodrigo Freitas Silva e Simone Dornelas Costa, por aceitarem avaliar e contribuir com esse trabalho.

A todos os professores do Departamento de Computação, por fazerem parte de minha vida nesses últimos anos.

RESUMO

Na atualidade é perceptível a expansão no uso de telefonia móvel. Com isso, a criação de aplicações móveis, que envolve diversos cenários, também se enquadra nesta realidade. Vários estudos vêm possibilitando o avanço das tecnologias, permitindo assim, oferecer diversos serviços aos usuários de telefonia móvel. Este trabalho teve como objetivo o desenvolvimento de um aplicativo que possa auxiliar os usuários a otimizar um problema que afeta diretamente diversas pessoas no Brasil, que são os elevados gastos em viagens, principalmente por causa dos altos preços dos combustíveis. Para isso, foi implementado uma aplicação para a plataforma móvel Android que auxiliará os usuários a elaborarem rotas entre dois pontos em um mapa e assim criar um plano de abastecimento ao longo desta rota. O aplicativo irá informar qual o melhor local e a quantidade de combustível a ser abastecida durante a viagem.

Palavras-Chave: Plataforma Móvel, Postos de Combustíveis, Otimização, Aplicativo Android.

SUMÁRIO

1. INTRODUÇÃO 8

1.1. O Problema e Sua Importância 9

2. OBJETIVOS 10

2.1. Objetivo Geral 10

2.2. Objetivos Específicos 10

3. REVISÃO DE LITERATURA 11

3.1. Carga tributária e Consumo de Combustíveis 11

3.2. Otimização de Rotas 12

3.3. Heurísticas 12

3.4. Teoria do Grafos 13

3.4.1. Problemas de Caminho Mínimo e o Algoritmo de Dijkstra 13

3.5. Sistema Operacional Android 14

3.5.1. Kernel Linux 15

3.5.2. Libraries e Android Runtime 16

3.5.3. Aplication Framework 16

3.5.4. Aplications 16

3.5.5. Estrutura de um Projeto Android 17

3.5.5.1. Activities 17

3.5.5.2. Services 18

3.5.5.3. Broadcast Receivers 19

3.5.5.4. Content Providers 19

3.5.6. Arquivos XML e o Andorid 19

3.6. Computação em Nuvem e Web Services 20

3.7. Goolge Maps API 22

3.7.1. Edição de Mapas 23

3.7.2. Criação de Rotas 23

3.7.3. Busca por locais 24

3.8. Apontador API 25

3.9. Android e Aplicativos 25

3.9.1. Aplicativos e a Google Maps API 26

4. METODOLOGIA 28

4.1. Ferramentas 28

4.1.1. Linguagem de Programação 29

4.2. Métodos 29
4.2.1. Processo de Desenvolvimento 29
4.2.2. Google Maps 30
4.2.3. Criação de Rotas 31
4.2.4. Busca por Postos de Combustível 33
4.2.5. Versão Final 34
5. RESULTADOS OBTIDOS 37
6. CONCLUSÃO 43
7. REFERÊNCIAS BIBLIOGRÁFICAS 44

1. INTRODUÇÃO

Com o advento da tecnologia móvel, cada vez mais os dispositivos móveis fazem parte da vida cotidiana das pessoas, conforme pode ser observado nos dados da Fling (2009). Dessa forma, o mercado de celulares vem crescendo cada vez mais. Segundo Lecheta (2013) estudos mostram que atualmente mais de 3 bilhões de pessoas possuem aparelhos celulares, o que corresponde a quase metade da população mundial. Atualmente tanto usuários comuns quanto empresas buscam incorporar aplicações móveis a seu cotidiano para agilizar negócios ou resolver problemas rotineiros. Seja para aumentar os lucros ou apenas para facilitar a rotina diária, os celulares e smartphones podem ocupar um importante espaço em um mundo onde a palavra “mobilidade” se torna cada vez mais importante.

Sendo assim, com o recente aumento na importância dos celulares, dados da International Data Corporation (IDC, 2013) afirmam que cerca de 78% dos aparelhos do mercado estão equipados com o Sistema Operacional Android, gerando assim uma grande demanda por aplicações móveis compatíveis com este sistema.

Existem diversas aplicações, com os mais variados fins, e todas elas estão disponíveis para os usuários na loja online da Google, a Google play, muitas delas tem seu foco em ajudar na resolução de problemas diários, como por exemplo: Aplicativos para resolução de problemas matemáticos, despertadores, agendas, mensageiros, calculadoras e outros. No entanto, além de resolver problemas, os aplicativos também podem ser usados para ajudar a otimizar a resolução de problemas. Esses aplicativos além de resolver um determinado problema, fornecem soluções que diminuem os custos ou maximizam os lucros e benefícios do usuário.

Assim, como exemplo desses aplicativos estão Dash (2015), que utiliza informações do GPS (sigla em inglês para Sistema de Posicionamento Global), combinadas com os serviços do Google Maps para informar rotas entre o local atual do dispositivo e um ponto que é buscado pelo usuário, assim como permite a busca por postos de gasolina, oficinas mecânicas e outras facilidades relacionadas à automóveis que estão ao redor da localidade atual do usuário; e o Runtastic (2015), que pode monitorar os gastos calóricos do usuário em uma caminhada ou corrida, através de cálculos feitos com a distância percorrida e tempo da atividade, que são monitorados com o auxilio dos serviços do Google Maps.

1.1. O Problema e sua Importância

Um problema que afeta muitos brasileiros atualmente é o elevado custo da locomoção automobilística, visto que o preço de combustíveis vem passando por recentes aumentos, conforme observado em Noticias Automotivas (2015), o que também pode ser observado em CNT (2015). De acordo com Minaspetro (2015) os combustíveis tiveram um aumento de 7,96% no mês de novembro de 2015. Essa sequência de aumentos segue uma tendência contrária ao que está ocorrendo no cenário internacional (G1, 2015). Tal fato faz com que os motoristas procurem novas maneiras de economizar ao abastecer o automóvel. Sendo assim, torna-se necessário encontrar uma forma para economizar os gastos com combustíveis, o que pode ser alcançado com um aplicativo móvel para otimização da resolução deste problema, visto que o recente estudo *adReaction* realizado pela Millward Brown (2014) mostra que os brasileiros passam cerca de 149 minutos por dia em frente a seus smartphones, tempo que é superior ao gasto em frente da televisão, que é de 113 minutos, e também do gasto em frente ao computador, que é de 146 minutos. Vale ressaltar também, que atualmente toda pessoa que tem um celular ou smartphone está sempre com o dispositivo ao alcance de suas mãos, ou seja, o aplicativo móvel, pode ser acessado à qualquer momento e sempre que necessário.

Então, um aplicativo móvel que otimiza, ou seja diminui, os custos com combustíveis, se torna importante para os mais diversos usuários, em viagens de longa ou curta distância, já que é possível economizar até 7% do dinheiro gasto se for feita uma pesquisa antecipada sobre os menores preços de combustível no caminho (UOL, 2014). Assim, a criação de uma aplicação móvel que possa otimizar o gasto de combustível em viagens é de grande relevância e por isso o presente trabalho busca o desenvolvimento de um aplicativo para a otimização deste problema. Tal aplicativo, será criado para a plataforma Android, que é atualmente a plataforma mais utilizada no Brasil, conforme pode ser observado em TechTudo (2012).

2. OBJETIVOS

2.1. Objetivo Geral

Desenvolver um aplicativo que seja capaz de determinar, em uma rota, o plano de abastecimento de um automóvel, ou seja, em quais locais o motorista deve parar para abastecer seu carro ao longo da viagem, levando em conta os preços de combustível em todos os postos dentro desta rota, e calcular o preço estimado de toda a viagem.

2.2. Objetivos Específicos

- Projetar a versão inicial do aplicativo com as funcionalidades do *Google Maps*;
- Desenvolver versão inicial do aplicativo para verificação da rota;
- Prototipar versão do aplicativo com informações sobre os postos de combustível;
- Desenvolver versão final do aplicativo com todas as funcionalidades;
- Testar e validar resultados obtidos.

3. REVISÃO DE LITERATURA

Neste trabalho é proposto o desenvolvimento de um aplicativo de otimização do custo do gasto de combustível em viagens, para isso abaixo são apresentadas informações e materiais correlatos.

3.1. Carga Tributária e Consumo de Combustíveis

De acordo com Maciel (2011) uma grande parte do preço de combustíveis no Brasil ocorre por conta da taxa tributária que incide sobre os mesmos, tal taxa tributária chega a ser em média, aproximadamente, 38% do preço final de todos combustíveis. Considerando como exemplo a gasolina produzida pela Petrobrás, é possível perceber a quantidade de tributos que incidem sobre a mesma: 17% de seu preço é referente à distribuição e revenda, 12% ao custo do Etanol Anidro, 28% deve-se ao ICMS, 11% é referente a outros impostos (CIDE, PIS/PASEP e CONFINS) e somente 32% é de realização da Petrobras (Petrobrás, 2015). Ao observarmos as porcentagens citadas, podemos perceber que, realmente, uma quantia significativa do preço da gasolina se deve à impostos.

Dito isso, é importante ressaltar, que atualmente o Brasil vem passando por momento de aumento das cargas tributárias, onde, de acordo com a Gazeta do Povo (2015) este ano chega a ser cotado como o ano com a maior carga tributária da história, e a taxa de tributos incidentes nos combustíveis também sofre aumentos (Economia UOL, 2015). Outro agravante para os recentes aumentos está ligado ao cenário de inflação vivido pelo Brasil (Noticias R7, 2015).

Visto que o preço de combustíveis está em alta, os motoristas vêm cada dia mais buscando formas alternativas de tentar economizar ao saírem com seus carros (Ric Mais, 2015). Segundo a Federação Nacional do Comércio de Combustíveis e de Lubrificantes (Fecombustíveis, 2015) os preços de combustíveis vêm assustando alguns consumidores, fazendo-os assim buscar pelo melhor posto de combustível, com o menor preço, antes de abastecer. Contudo, uma grande preocupação dos motoristas, está nos gastos em viagens de média e longa distância, pois é nesse tipo de viagem em que o consumo de combustível se torna elevado e consequentemente o custo também se eleva. Assim, é de suma importância um bom

planejamento da viagem, visto que um bom plano do abastecimento nos postos mais baratos ao longo do trajeto pode gerar uma economia de até 7% (Pense Carros, 2014).

3.2. Otimização de Rotas

Segundo Pinheiro (2013) uma rota é uma sequência de pontos específicos que o veículo deve percorrer ordenadamente iniciando e terminando em um ponto especifico, pré-determinado. Sendo assim, ao levar em consideração o cenário brasileiro, é de suma importância otimizar a rota a ser seguida em uma viagem, pois quanto menos quilômetros forem percorridos, menor será o consumo de combustível, gerando assim uma economia no gasto total da viagem. Assim, existem diversos algoritmos e técnicas para se encontrar a menor rota entre dois locais. Uma grande parte desses algoritmos e técnicas usam o conceito de heurísticas (Malaquias, 2006) ou se baseiam na Teoria dos Grafos (Wosley, 1998). Além disso, existe a possibilidade de criação de rotas entre dois locais por meio da Google Maps API. Tais métodos serão abordados com maior ênfase a seguir.

3.3. Heurísticas

Uma heurística pode ser vista como uma técnica baseada em processos intuitivos, buscando uma boa solução para algum problema com um custo computacional aceitável. Estas técnicas não garantem uma solução ótima para um determinado problema e também não permitem uma análise de quão perto a solução dada está da ótima, porém, se montadas de forma correta, podem trazer bons resultados em tempos muito inferiores aos métodos lineares. Tais métodos são sempre aplicados a problemas muito restritos, assim gerando heurísticas muito especificas para aquele tipo de situação, deste modo um método criado para resolver um dado problema, não é viável para resolver questões mais amplas e abrangentes (Wosley, 1998).

3.4. Teoria do Grafos

Em uma definição bem simples, grafos são pares ordenados, da forma G = (V,A), tais que V é um conjunto de Vértices e A é um conjunto de Arestas que conectam pares de Vértices. De acordo com Netto (2006) a Teoria dos Grafos é a disciplina que estuda as propriedades e principais técnicas e características dessa modelagem. Grafos podem ser utilizados para resolver inúmeros tipos problemas, e dentre eles estão os problemas de caminho mínimo, que consistem em achar a melhor rota, ou menor caminho, entre dois locais.

3.4.1. Problemas de Caminho Mínimo e o Algoritmo de Dijkstra

Como demonstrou Gersting (2004), o problema de Caminho Mínimo consiste basicamente em achar o caminho de menor custo entre dois vértices *x* e *y* de um grafo, dado que as arestas do grafo possuem um determinado valor (custo). Assim, para a resolução de tais problemas podem ser utilizados diversos algoritmos que retornam o melhor caminho (menor custo) entre os vértices *x* e *y*, e dentre eles pode-se citar o algoritmo de Dijkstra que é o mais conhecido e robusto para a resolução deste tipo de tarefa. Gersting (2014) ainda diz que tal algoritmo consiste, basicamente, na escolha de vértice *x* como raiz da busca, assim calculando o custo mínimo deste vértice para todos os demais vértices do grafo. Para isso, ele parte de uma estimativa inicial para o custo mínimo e vai sucessivamente ajustando esta estimativa. Ele considera que um vértice estará fechado quando já tiver sido obtido um caminho de custo mínimo do vértice tomado como raiz da busca até ele. Caso contrário ele é considerado aberto. Assim em cada iteração é adicionado o vértice aberto que possui a menor distância em relação a *x*, após a análise de todos os vértices adjacentes que podem gerar um menor caminho.

Na Figura 1, pode-se observar o funcionamento, passo a passo, do algoritmo de Dijkstra.

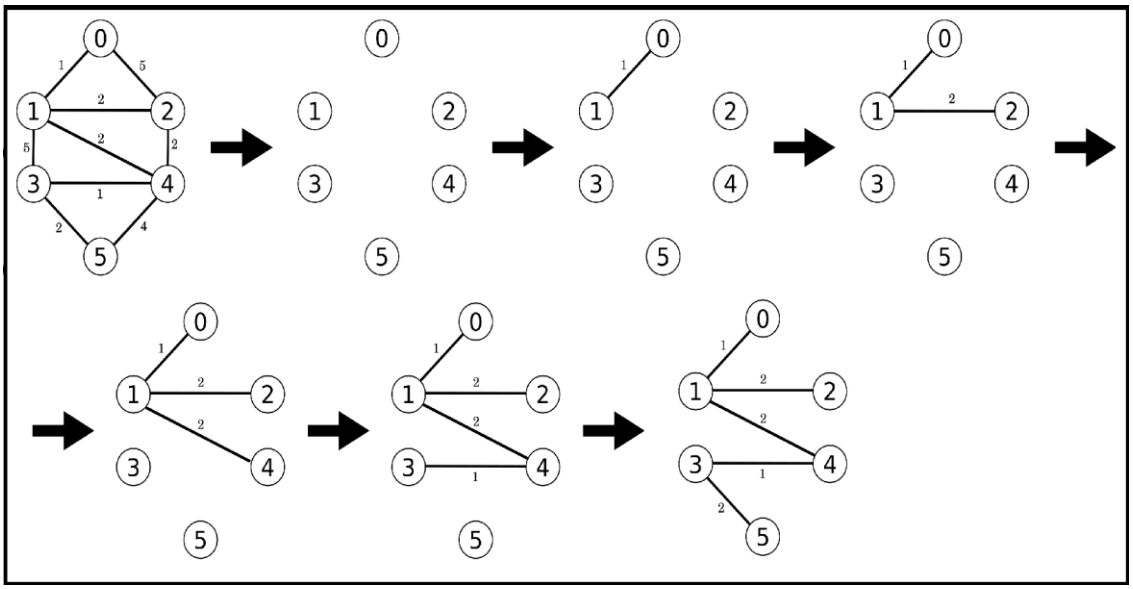

Figura 1 – Funcionamento do Algoritmo de Dijkstra.

Fonte: adaptada de Ziviani (2004)

3.5. Sistema Operacional Android

Silva (2010) diz que o Android é uma plataforma desenvolvida pela Google em parceria com empresas da *Open Handset Alliance* (OHA), como a *Samsung*, *Motorola*, *HTC*, *Intel*, *LG* e outras, baseada no Linux e com foco em dispositivos móveis, como *smartphones* e *tablets*. O Android foi desenvolvido com o intuito de disponibilizar para o mercado uma plataforma padrão para dispositivos móveis, independente de *hardware* ou fabricante, para assim garantir ao usuário o poder de escolha de características do aparelho que lhe são mais vantajosas..

De acordo com Freitas (2012) o Android possui uma arquitetura baseada em 4 níveis interdependentes, que podem ser observados na Figura 2 e serão descritos em seguida.

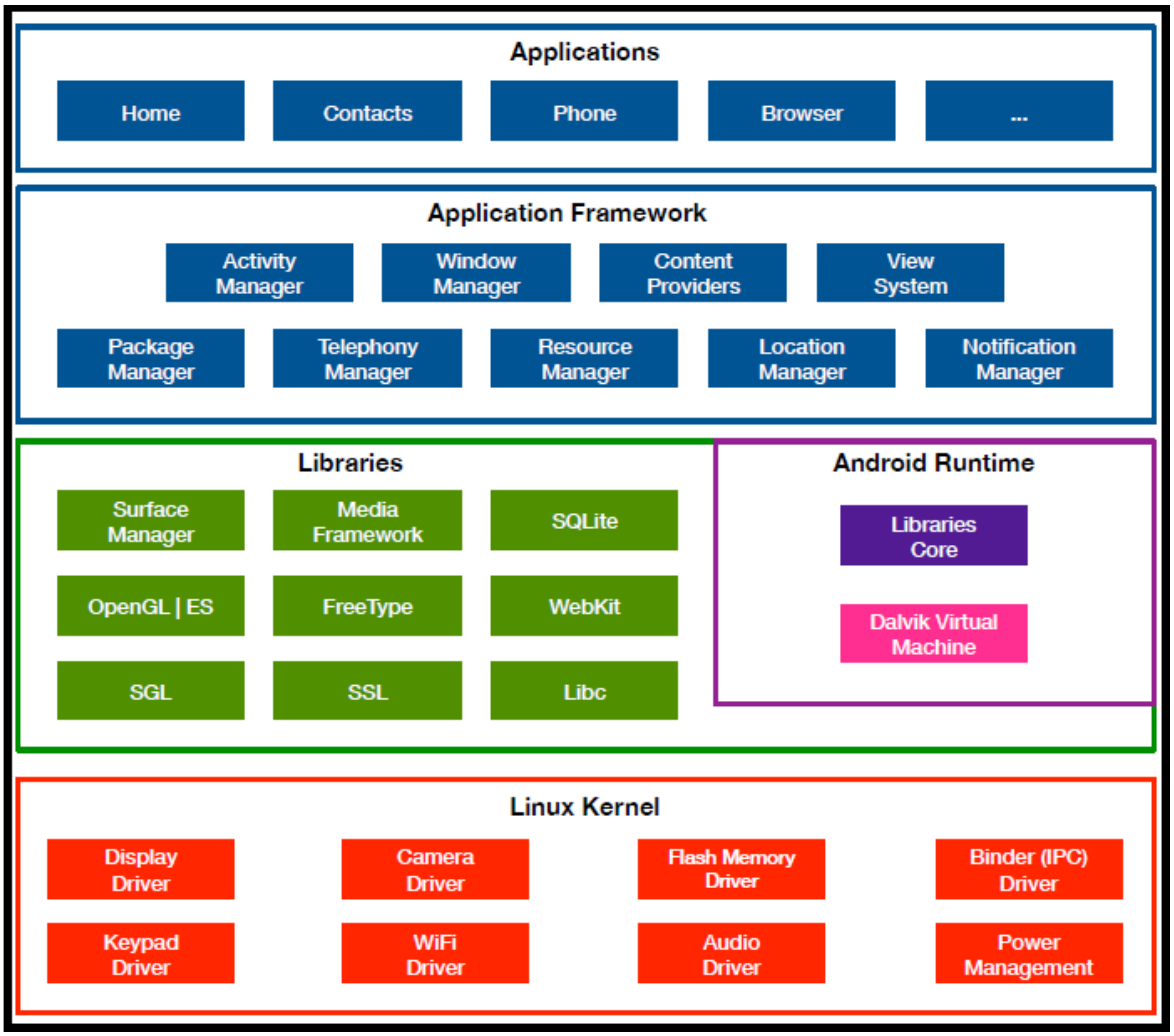

Figura 2 – Arquitetura Android.

Fonte: Freitas (2012)

3.5.1. Kernel Linux

É o primeiro nível da arquitetura e é baseado em uma modificação do *kernel* 2.6 do Linux. Este nível é o responsável pela inicialização do sistema, gerenciamento de memória, processos, *threads* em execução e energia do dispositivo. Todos os drivers responsáveis pelas comunicações entre recursos e *hardware* se encontram nesse nível (Freitas, 2012; Pereira & Silva, 2009).

3.5.2. Libraries e Android Runtime

Este é o segundo nível da arquitetura, e é composto por duas partes: *Libraries* que são bibliotecas nativas do Android, escritas em C/C++ que executam diretamente no Linux. Os aplicativos desenvolvidos não tem acesso direto à essas bibliotecas, tal acesso é abstraído pelo nível superior *Aplication Framework*; e *Android Runtime* que apresenta uma subdivisão em duas partes: *Core Libraries* que são bibliotecas do Java e compreendem praticamente todas as bibliotecas do Java SE (ferramenta de desenvolvimento para a plataforma Java); e *Dalvik Virtual Machine* que é uma maquina virtual (MV) do Android, onde os aplicativos são executados. Esta MV não é a mesma que executa aplicações Java normais, ela executa arquivos .DEX (*Dalvik Executable*) otimizados para o melhor desempenho em dispositivos com recursos de processamento e memória limitados. Estes arquivos são classes compiladas Java e transformadas em arquivos .DEX (Freitas, 2012; Pereira & Silva, 2009).

3.5.3. Aplication Framework

Corresponde ao terceiro nível da arquitetura e compreendem as APIs (Aplication Programing Interface, em português: Interface de programação de aplicativos) do Android que são usadas no desenvolvimento de aplicativos. Esta camada abstrai o acesso às bibliotecas da camada de *Libraries* (Freitas, 2012; Pereira & Silva, 2009).

3.5.4. Aplications

Esta é a quarta e ultima camada da arquitetura e também a superior, compreende os aplicativos escritos em Java, tanto aplicativos nativos do Android, quanto aplicativos desenvolvidos por terceiros (Freitas, 2012; Pereira & Silva, 2009).

3.5.5. Estrutura de um Projeto Android

Freitas (2012) diz que um projeto Android, apesar de usar Java para sua codificação tem uma estrutura um pouco diferente de projetos convencionais para essa linguagem, se estruturando em quatro componentes que podem ser instanciados e executados pelo sistema sempre que necessário, os quais podem ser observados na Figura 3 e serão explicados a seguir.

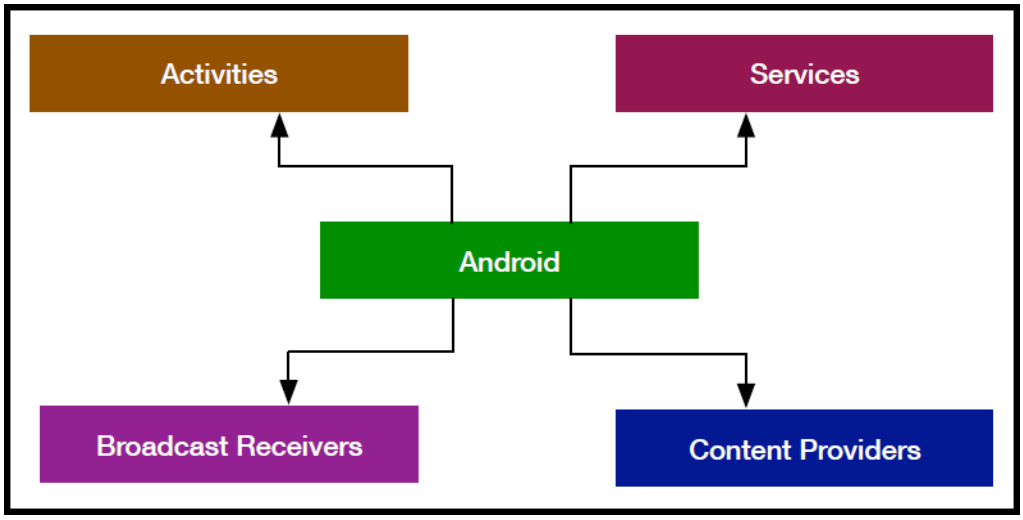

Figura 3 – Estrutura de um Projeto Andorid.

Fonte: Freitas (2012)

3.5.5.1. Activities

Pereira e Silva (2009) dizem que a *Activity* (Atividade) é o mais utilizado de todos os componentes e possui uma interface composta por *Views* (Telas que respondem a eventos previamente programados). E cada *Activity* possui um ciclo de vida específico, baseado nas seguintes funções:

- *OnCrate()*: método chamado quando a atividade é inicialmente criada;
- *OnStart()*: instanciado quando a atividade torna-se visível para o usuário;
- *OnResume()*: é o topo da pilha de atividade, chamado quando se inicia a interação com o usuário;
- *OnPause()*: chamado quando o sistema começa uma nova atividade (como uma chamada recebida).

- *OnStop()*: instanciado quando a atividade não está mais sendo utilizada.
- *OnDestroy()*: destrói e pode ser chamado quando a atividade terminar, ou quando o sistema necessitar de memória.
- *OnFreeze()*: possibilita salvar o estado atual da atividade quando ela está na iminência de ser suspensa.

Assim que uma atividade é iniciada ela é colocada no topo da "pilha de atividades" e se torna a atividade em execução. Pode-se observar a seguir, na Figura 4, um esquema com o ciclo de vida de uma *Activity*.

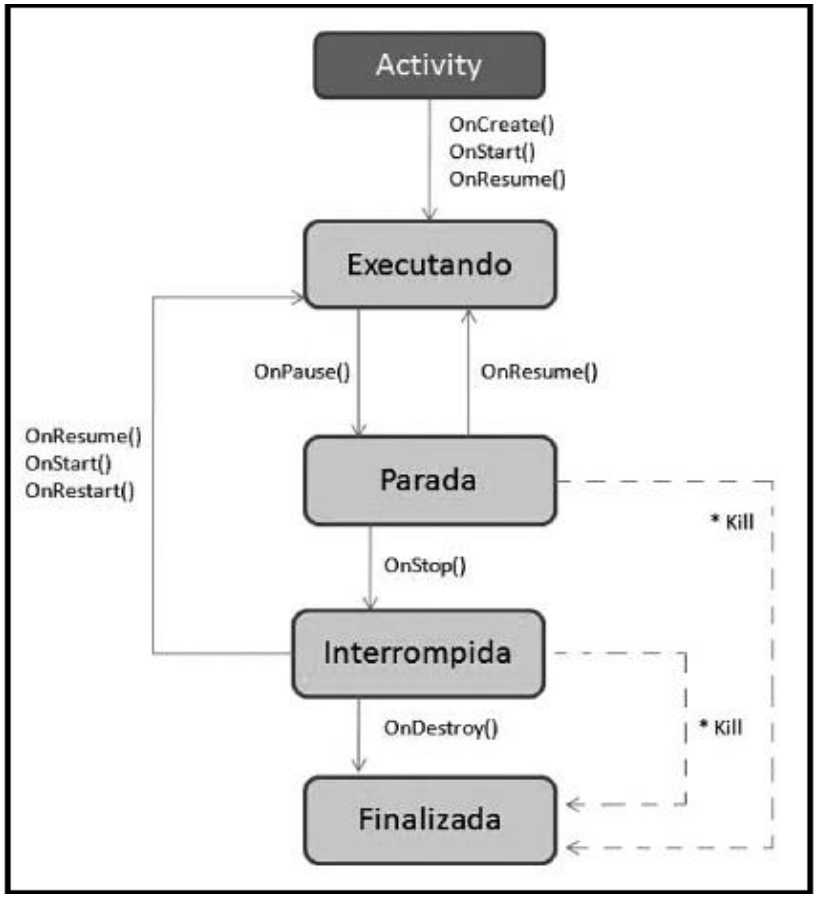

Figura 4 – Ciclo de Vida de uma Activity

Fonte: Pereira e Silva (2009)

3.5.5.2. Services

Os *Services* são códigos executados em *background* (segundo plano) e não são interrompidos pela troca de atividades. Ao contrario das *Activities* que tem ciclo

de vida próprio e curto, os *Services* mantém os serviços ativos até receberem uma outra ordem (Pereira & Silva, 2009).

3.5.5.3. Broadcast Receivers

Segundo Freitas (2012) um *Broadcast Receiver* é um componente que responde a determinados anúncios enviados pelo sistema. O Android envia mensagens para todo o sistema quando determinado evento ocorre, estas mensagens são chamadas de *broadcasts*, e o componente responsável por recebe-las e responde-las são os *Broadcast Receivers*. Como por exemplo, quando a bateria está fraca, o Android manda um *broadcast* avisando aos aplicativos que tem interesse sobre essa informação, para que eles possam executar uma determinada ação a partir desta informação.

3.5.5.4. Content Providers

Fornece uma forma de compartilhamento de informações entre aplicativos de forma segura. Através dos *Content Providers* pode-se permitir que aplicações armazenem, modifiquem e recuperem dados em um repositório compartilhado (Pereira & Silva, 2009).

3.5.6. Arquivos XML e o Andorid

O XML, do Inglês *eXtensible Markup Language*, é uma linguagem de marcação recomendada pela W3C (*World Wide Web Consortium*), que é a principal empresa de padronização da *World Wide Web* (WWW), para a criação de documentos que contenham dados com uma organização hierárquica, visto que o XML é composto por unidades chamadas entidades, que são compostas por dados. Todas as entidades tem algum conteúdo, e são identificadas por um nome, elas são compostas por uma *tag* (etiqueta) de inicio e uma *tag* de fim e entre essas *tags* podem ser encontradas outras entidades ou dados referentes à entidade em questão. O XML foi criado para ser usado diretamente pela internet, facilitar a comunicação entre diferentes aplicações e para ser de fácil entendimento e criação.

Portanto, uma de suas principais características é a portabilidade, pois aplicações que estão em sistemas operacionais diferentes, são codificadas em linguagens diferentes ou tem estruturas totalmente incompatíveis podem se comunicar utilizando arquivos XML (Bray et al., 2006). Na Figura 5 é apresentado um exemplo de arquivo XML e suas estruturas.

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

Figura 5 – Exemplo de Arquivo XML.

Fonte: XML (2015)

Os arquivos XML são de suma importância para o Android, pois o sistema se baseia em uma associação de Java + XML, onde o Java fornece a codificação para funções e eventos, e o XML fornece a estrutura para o *layout* de *Views*, bem como várias definições referentes à aplicação. Além disso, várias aplicações Android fazem o uso de comunicação com a web, a qual é feita, muitas vezes, através da troca de arquivos XML (Silva, 2010).

3.6. Computação em Nuvem e Web Services

A computação em nuvem está se tornando uma das palavras chaves da indústria de TI. A nuvem é uma metáfora para a Internet ou infraestrutura de comunicação entre os componentes arquiteturais, baseada em uma abstração que oculta à complexidade da infraestrutura. Cada parte desta infraestrutura é provida como um serviço e, estes são normalmente alocados em centros de dados utilizando hardware compartilhado para computação e armazenamento (Buyya et al. 2009).

Tais dados que estão disponíveis na nuvem podem ser acessados através dos chamados *Web Services*, que de acordo com W3C (2015), providenciam meios

de interoperação entre diferentes aplicações cliente-servidor em diferentes plataformas, que se comunicam através da WWW, utilizando o *HyperText Transfer Protocol* (HTTP). As principais características destes serviços são sua interoperabilidade e extensibilidade. A W3C (2015) ainda diz que esses serviços tem a capacidade de diminuir a carga de processamento que acontece nos dispositivos (clientes) que os utilizam, pois enviam para servidores as informações necessárias para o processamento de um determinado problema, o qual faz o processamento pesado e responde em forma de um arquivo XML, por exemplo, que possui a resposta para o problema em questão.

A Figuras 5 contém um exemplo de como é a estrutura e como funciona um *Web Service.*

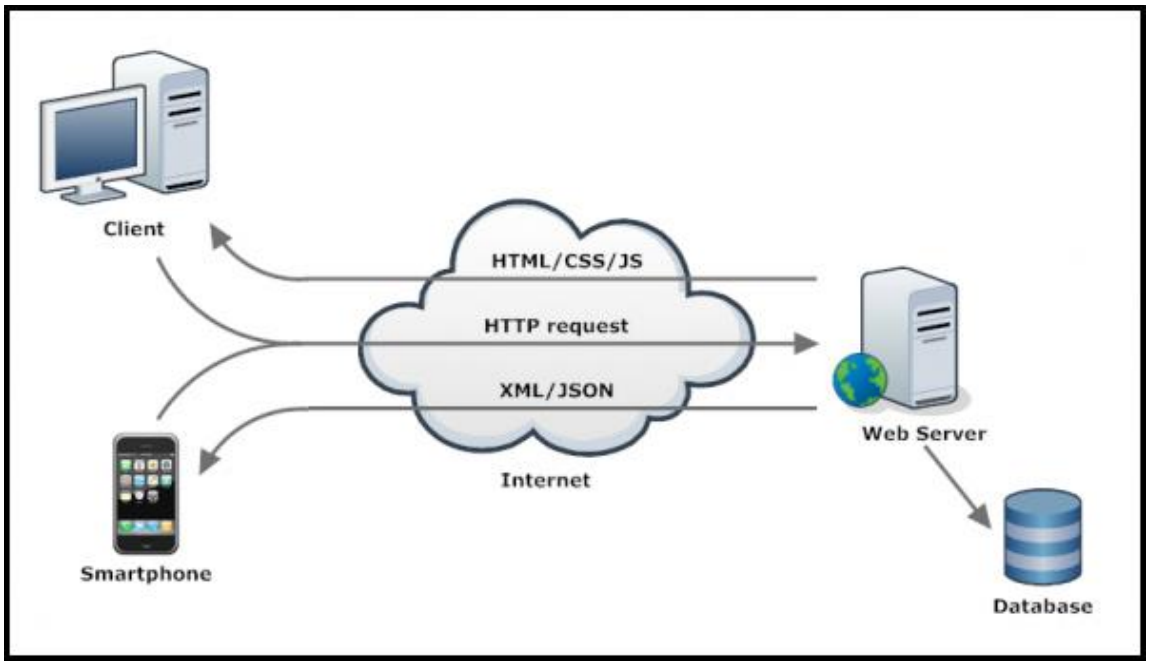

Figura 5 – Funcionamento de *Web Service*.

Fonte: Web Service (2014)

A Figura 6 demonstra um exemplo de uma requisição e parte de sua resposta XML para uma aplicação do Google *Maps API*.

Requisição

http://maps.google.es/maps/api/directions/xml?origin=Av Ipiranga 3200, Porto Alegre – RS&destination=Av Joao Pessoa, Porto Alegre – RS&sensor=false

Resposta

```
<?xml version="1.0" encoding="UTF-8"?>
<DirectionsResponse>
<status>OK</status>
<route>
<summary>Av. Ipiranga</summary>
<leg>
<step>
<travel mode>DRIVING</travel mode>
<start location>
<lat>-30.0465300</lat>
<lng>-51.1975700</lng>
</start location>
<end location>
<lat>-30.0467000</lat>
<lng>-51.2119800</lng>
</end location>
```

Figura 6 – Exemplo de Requisição e Resposta de um Web Service.

Fonte: Adaptada de *Google Directions e Elevation API* (2010)

3.7. Google Maps API

Interface de Programação de Aplicativos, ou simplesmente API, é a forma que temos de nos comunicarmos com as funções pré-programadas e pré-definidas por um fornecedor, podendo ser utilizada a internet para fazer a conexão entre o cliente (usuário) e o servidor (fornecedor) através de *Web Services*. Uma das APIs mais utilizadas é a Google Maps API desenvolvido pela Google. Tal API, nos permite a manipulação de mapas, criação de locais, pesquisas por estabelecimentos, geração de rotas, dentre outros. Atualmente tal interface é a mais utilizada no mundo, com cerca de 55 milhões de downloads (Google Maps API, 2015).

Neste trabalho, serão usadas algumas funcionalidades dessa API.

3.7.1. Edição de Mapas

A edição de mapas dá a possibilidade de criação de novos mapas, assim tornando possível a inserção de pontos personalizados, que podem conter informações úteis relacionadas ao local de referência. Isso possibilita a demarcação de locais dentro de uma rota pré-determinada (Google Developer, 2015).

Pode-se observar na Figura 7 um exemplo de mapa editado com a Google *Maps* API para Android.

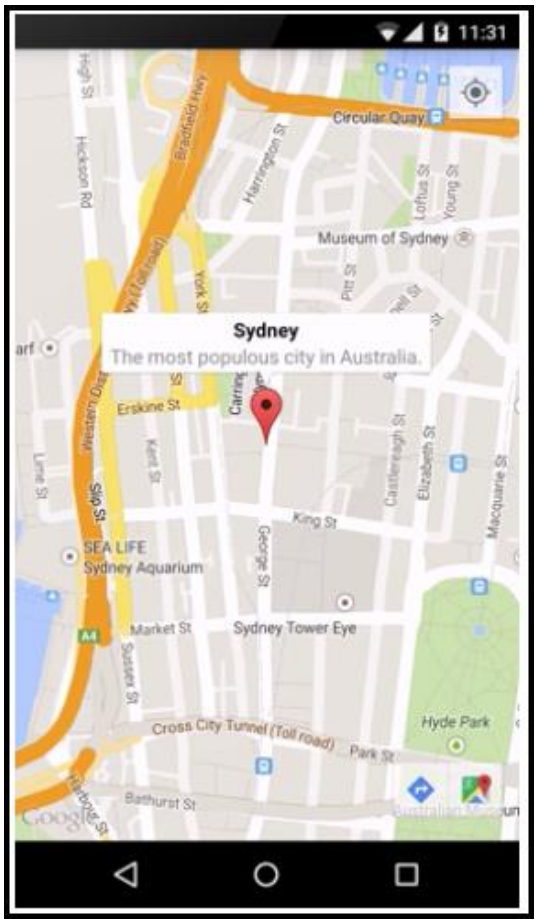

Figura 7 – Exemplo de Mapa Editado com a API do Google Maps.

Fonte: Google Maps API (2015)

3.7.2. Criação de Rotas

A criação de rotas ocorre através da demarcação de um ponto de partida, pontos intermediários, que podem ou não ser incluídos na rota, e um ponto de chegada. Assim a API calculará a melhor rota, sendo que é permitido adicionar

restrições às rotas, como não passar por pedágios, pontes, estradas não pavimentadas, dentre outros (Google Developer, 2015).

Na Figura 8 encontra-se um exemplo de criação de uma rota personalizada em uma aplicação para web.

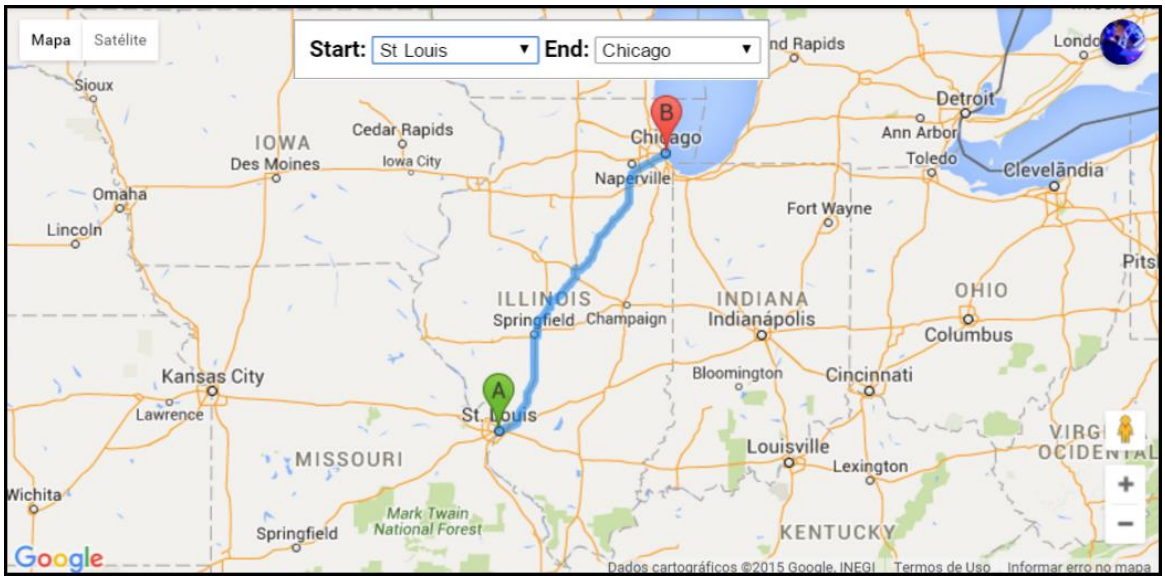

Figura 8 – Exemplo de Criação de uma Rota Personalizada.

Fonte: Google Directions (2015)

3.7.3. Busca por Locais

A API do *Google Maps* através de uma de suas API's auxiliares, *Google Places*, permite consultar informações de locais em uma variedade de categorias, como estabelecimentos comerciais, pontos de interesse proeminentes, localizações geográficas, dentre outros. Pode ser feita uma busca através de um tipo de localidade ou por localidades próximas ao usuário. Uma *Place Search* (Busca por Local) retorna uma lista de locais relacionadas com a busca realizada e cada local dessa lista possui um conjunto de informações, tais como: nome do local; avaliações de usuários; faixa de preço (que varia de 0, gratís, a 4, muito caro); horário de funcionamento; dentre outras.

A Figura 9 mostra os resultados de um exemplo de *Place Search* no *Google Maps* para web, onde foram pesquisados postos de combustível na região metropolitana de Vitória-ES.

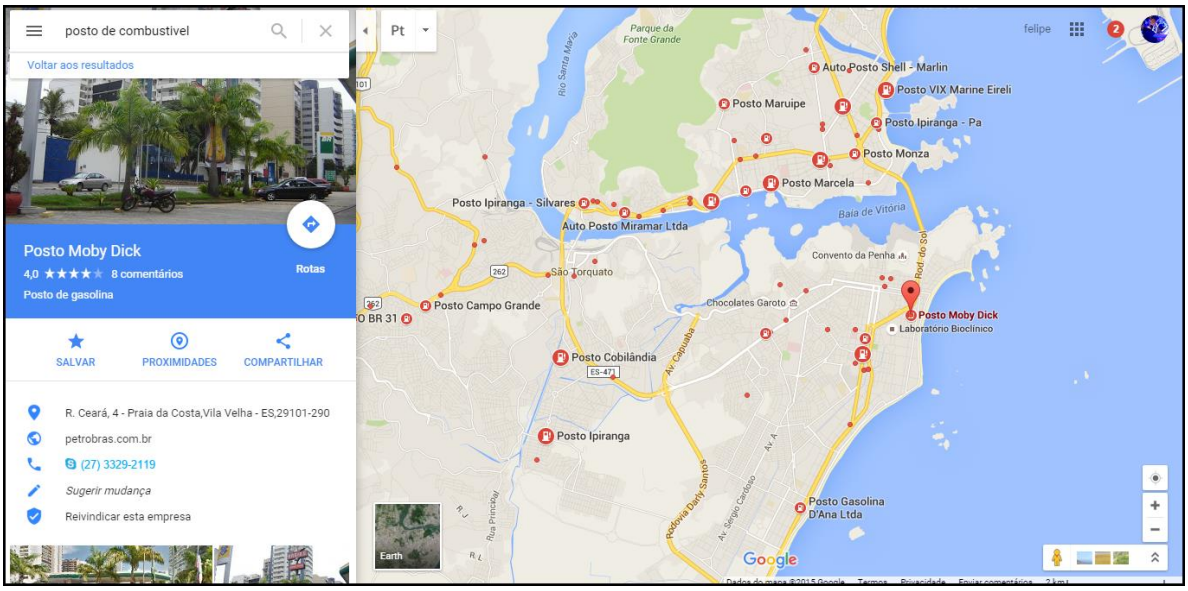

Figura 9 – Exemplo de Busca por Postos de Combustível.

Fonte: Places (2015)

3.8. Apontador API

Uma outra API, bastante utilizada, é a Apontador API, que permite fazer a busca de locais pré-estabelecidos em um banco de dados que se encontra presente na nuvem, retornando várias informações sobre determinado local, como: preço de produtos; avaliação de usuários; fotos do local; dentre outras informações pertinentes. Para tal busca, é necessário passar como informação uma localização inicial, um parâmetro do que se está procurando (filtragem de locais), e um raio de busca, que determina quão longe do ponto inicial devem ser feitas as buscas (Apontador, 2015). Uma recente atualização na API apontador trouxe novas funcionalidades e mais rapidez e confiabilidade nas respostas dadas pela mesma, assim como a retirada de alguns serviços como a informação do preço de produtos (Apontador V2, 2015).

3.9. Android e Aplicativos

Segundo recentes dados do IDC, que é uma empresa especializada na análise do mercado tecnológico, em 2013 foram vendidos aproximadamente 1 bilhão de smartphones, sendo que deste montante, cerca de 78% dos aparelhos está

equipado com o Sistema Operacional Android. Assim, como consequência, temos a crescente demanda por aplicações que sejam executados nestes dispositivos móveis. A popularidade é tamanha que segundo a *AppBrain* (2014) o número de aplicativos disponíveis na Google Play, que é a loja de aplicativos oficial da Google, ultrapassou 1 milhão. Este grande número de aplicativos atende a diversas demandas, contendo aplicações que resolvem a variados tipos de problemas, oferecendo soluções aos usuários para questões simples do dia a dia e também para outras mais complexas. Podemos citar alguns aplicativos famosos, por serem usados por muitas pessoas, como: Whatsapp, que proporciona a troca de mensagens entre pessoas de forma simples e rápida (Whatsapp, 2014); Palco MP3, que permite ao usuário ouvir músicas baseadas em seu gosto musical, buscando por gênero ou por artistas de seu gosto (Palco MP3, 2014); Facebook para Android, o aplicativo contém todas as funcionalidades do Facebook acessado de um computador, porém é otimizado para o uso em dispositivos móveis (Facebook, 2014); Dubsmash, é um aplicativo que permite a dublagem de diversos sons, fazendo vídeos dos mais diversos tipos (Dubsmash, 2015); Instagram, rede social de fotos, que permite a postagem e edição básica de fotos instantaneamente (Instagram, 2014); Sinesp Cidadão, aplicativo de utilidade pública, que permite a consulta de informações da base de dados do Departamento Nacional de Transito (Sinesp, 2014); Além dos mais diversos jogos (Jogos, 2015), aplicativos de entretenimento (Android Entretenimento, 2015), aplicações matemáticas e cientificas (Matematicascercanas, 2015), dentre outros.

3.9.1. Aplicativos e a Google Maps API

O Google Maps também possui uma API especifica para dispositivos móveis, a qual beneficia vários desenvolvedores de aplicativos, pois facilita a confecção de aplicações com suas mais variadas funções e bibliotecas (Geoambiente, 2014). Sendo assim, podemos citar alguns aplicativos que utilizam as funcionalidades desta API: MyTaxi, aplicativo móvel que oferece a solicitação de taxis online (MyTaxi, 2014); Gazo, que permite ao usuário a busca pelo posto de gasolina com o melhor preço em sua região (Gazo, 2014); Preços de Combustíveis, permite a busca por postos de combustível que estão ao redor da localidade atual do

dispositivo ou em algum ponto de terminado pelo usuário, assim como mostrar os preços de combustíveis destes postos pesquisados (Preços de Combustíveis, 2015); Gasnol, o aplicativo informa ao usuário qual o melhor combustível a ser usado em seu veículo, etanol ou gasolina, assim como a busca por postos de combustíveis que estão ao redor do dispositivo (Gasnol, 2015).

4. METODOLOGIA

Segundo Jung (2004), Silva e Menezes (2001) e Moresi (2003) as pesquisas podem ser classificadas quanto à sua natureza, forma de abordar o problema, objetivos e meios de investigação. Desta forma, de acordo com as definições e classificações dadas por estes autores, esta pesquisa se enquadra quanto a natureza como tecnológica, pois conhecimentos básicos e tecnologias existentes são empregados para a solução do problema, também classifica-se, quanto a forma de abordar o problema, como uma pesquisa qualitativa, visto que o estudo não pode ser quantificado em números. O presente estudo ainda pode ser classificado quanto aos objetivos como uma pesquisa exploratória, visto que se tentou aplicar a um problema hipotético teorias e conceitos previamente existentes, assim como, também podemos classificar o trabalho quanto aos meios de investigação, como um estudo de caso.

Visto isso, para a realização deste trabalho, que consistiu no desenvolvimento de um aplicativo para a plataforma móvel Android, que tem por finalidade a otimização do gasto de combustível, principalmente em viagens, foram utilizados métodos, ferramentas e conceitos, que serão abordados e descritos nos itens a seguir.

4.1. Ferramentas

A aplicação foi desenvolvida e codificada no Ambiente Integrado de Desenvolvimento (da sigla em inglês IDE) *Android Studio*, que é a IDE oficial para o desenvolvimento de aplicações para Android (Android Studio, 2014). Sendo assim, para o desenvolvimento de algumas das funcionalidades do aplicativo, foi utilizada a API do Goolge Maps e suas derivadas, que visam otimizar e facilitar o processo de codificação, visto que tal API fornece diversas facilidades e funcionalidades que foram de suma importância para o desenvolvimento do trabalho.

4.1.1. Linguagem de Programação

A linguagem escolhida para a codificação do projeto foi o Java, que é padrão para o desenvolvimento de aplicações para Android. Tal linguagem foi escolhida devido a sua modernidade, portabilidade e segurança. Além disso, segundo Anguish, Buk e Yacktman (2002), tal linguagem é ideal para o desenvolvimento de softwares que executam processos de longa duração em um servidor e que fazem uso da rede de computadores, para comunicação e de um banco de dados, para obtenção de informações. E por fim, o Java é uma linguagem de código aberto e amplamente difundida e utilizada, com cerca de 9 milhões de desenvolvedores em todo o mundo (Oracle, 2012).

4.2. Métodos

Nesta seção, serão abordados os métodos que foram utilizados na produção da aplicação proposta e também mostrado como é a arquitetura geral do aplicativo e como foram implementadas suas funcionalidades.

4.2.1. Processo de Desenvolvimento

O processo aplicado no desenvolvimento do aplicativo pode ser dividido em três etapas principais (desenvolvimento, teste, validação) conforme o diagrama de fluxo de atividades apresentado na Figura 11.

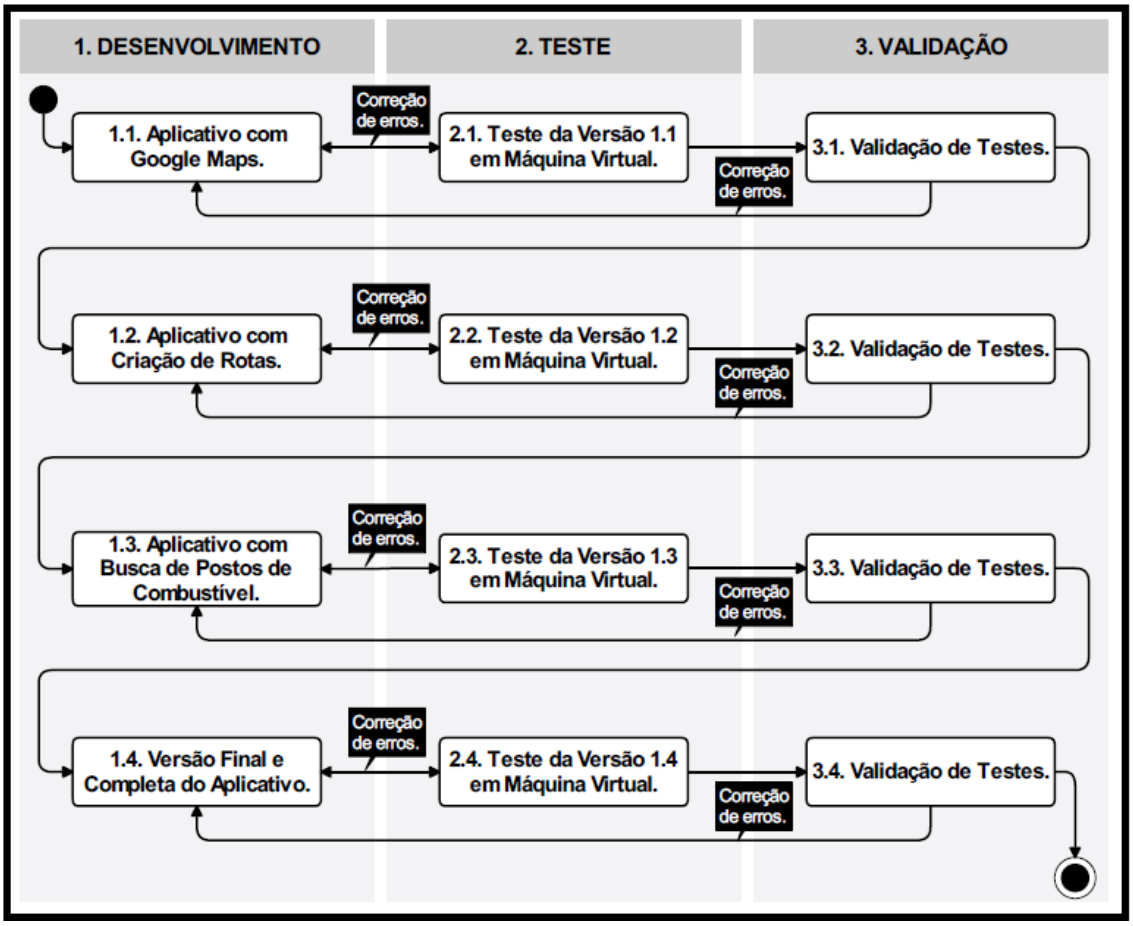

Figura 11 – Diagrama de Fluxo de Atividades do Projeto.

4.2.2. Google Maps

Ao se desenvolver um aplicativo que contem uma *Activity* com a utilização de um mapa fornecido pela Google, é necessário utilizar a API do Google Maps. Para tal, foi necessário o cadastro do aplicativo no *console* de desenvolvimento da Google, que é uma plataforma para a criação e gerenciamento de aplicativos que utilizam recursos fornecidos por suas API's. Assim a Google fornece uma chave identificadora única para o projeto criado, que é necessária para a autenticação do aplicativo nas requisições feitas à API's deste fornecedor (Developer, 2015). Após este cadastro, bastou-se criar um projeto no *Android Studio* e optar pela opção de *Maps Activity.* A partir disso foi adicionada a chave identificadora gerada ao projeto e feita pequenas configurações nas permissões de acessos à internet e localização do dispositivo.

Na Figura 12, pode-se observar o resultado do processo descrito.

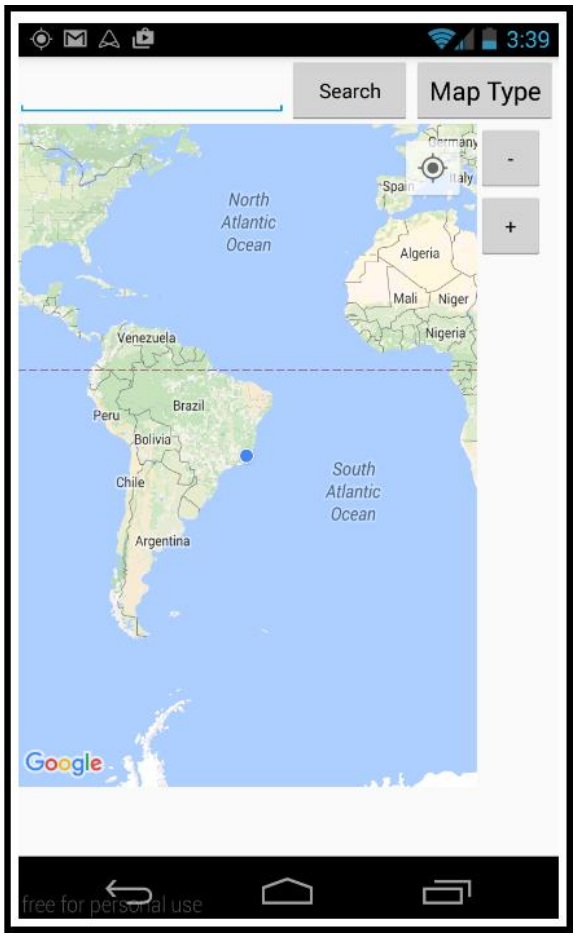

Figura 12 – Mapa Gerado com a Google Maps API

4.2.3. Criação de Rotas

Para traçar uma rota em um mapa gerado pela API do Google Maps são necessários um ponto de partida e um ponto de chegada, que foram obtidos através da latitude e longitude da localização atual do dispositivo (ponto de partida) e uma pesquisa à API, passando como parâmetro uma *string* contendo o nome de uma localidade e obtendo como resposta uma latitude e longitude referentes à localização do local pesquisado. Então estas latitudes e longitudes foram adicionadas a uma URL (sigla em inglês para *Uniform Resource Locator*) que faz uma requisição de rota entre estes pontos à API Google Directions (API auxiliar do Google Maps). Tal requisição pode ser vista na Figura 13, que contem uma solicitação de rota entre dois pontos.

```
https://maps.googleapis.com/maps/api/directi
ons/xml?origin= -20.7628,-
41.5324&destination=-20.2976178,-40.2957768
&sensor=false
```

Figura 13 – Exemplo de Requisição à API Google Directions

Após essa requisição, a API Google Directions fornece um arquivo XML, semelhante ao observado na Figura 6, que contem latitudes e longitudes de diversos pontos entre a partida e a chegada, além de informações de distância e duração estimada do percurso entre esses pontos. Então foram analisados estes dados e extraídas as informações pertinentes para que fosse possível a geração de uma rota, a qual foi desenhada no mapa gerado pela API do Google Maps. Assim, o resultado deste processo pode ser observado na Figura 14. Para cada solicitação de rota, esse processo é repetido com um novo ponto de chegada, e com um novo ponto de partida, caso o dispositivo não estiver mais no mesmo local das requisições anteriores.

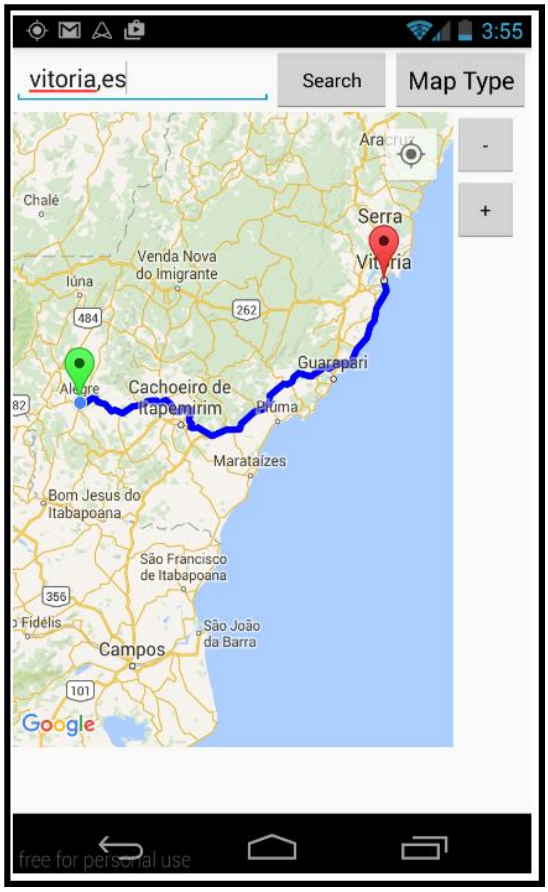

Figura 14 – Rota Gerada

4.2.4. Busca por Postos de Combustível

A busca por locais foi feita utilizando a API Google Places (API auxiliar do Google Maps) da seguinte maneira: foi feita uma requisição à API, através de uma URL, contendo uma longitude, um raio de busca dado em metros (para delimitar a busca a uma determinada distância) e uma *string* com a palavra "*gass_station*" (para filtrar a busca por postos de combustível). A Figura 15 contém um exemplo de URL gerada para uma busca de locais.

```
https://maps.googleapis.com/maps/api/place/n
earbysearch/xml?location=-20.2976178,-
40.2957768&radius=50000&types=gas_station&se
nsor=false&key=AIzaSyBxKc_A1yHF8Ebwg_1MBXbR9
PdSdQn2KVw
```

Figura 15 – Exemplo de Requisição à API Google Places

Após essa requisição, a API responde com um arquivo XML que contém as informações sobre os postos de combustível que estão ao redor do ponto informado e dentro do raio de busca. Assim, se fez necessária uma análise deste arquivo para a extração da latitude e longitude dos locais e de outras informações pertinentes como o nome do local, pare que assim fosse possível mostrar estes locais no mapa gerado pela API do Google Maps. Com isso, o resultado para esta rotina pode ser observado na Figura 16, e para cada busca diferente esse processo se repete.

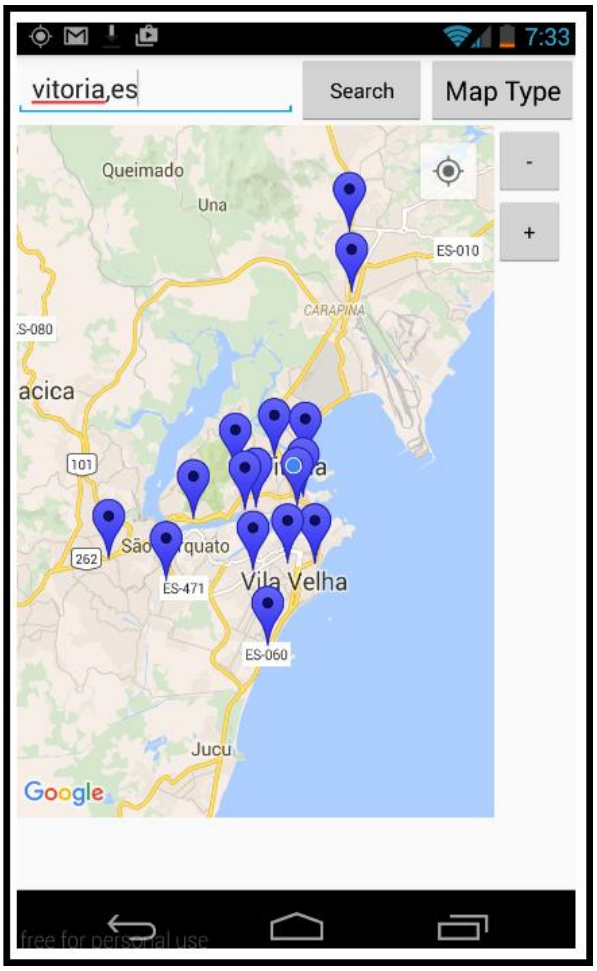

Figura 16 – Busca por Postos

4.2.5. Versão Final

Como pode ser observado na Figura 11, o processo de desenvolvimento do aplicativo foi incremental, ou seja, a cada etapa concluída foi agregada uma nova funcionalidade ao aplicativo. Para o desenvolvimento da versão final do aplicativo foi necessário o aperfeiçoamento das funções já presentes, e para isso foi criada uma nova *Activity* que possui uma nova *View* que serve como tela inicial, conforme pode ser observado na Figura 17, que contém 4 campos de inserção de dados referentes, respectivamente, à capacidade de armazenamento (em litros) do tanque de combustível, à quantidade de litros dentro do tanque, à quilometragem por litro e ao tipo de combustível utilizado pelo veículo do usuário; e um botão para iniciar o aplicativo. Após o preenchimento destes campos o usuário clica no botão de iniciar e as informações são salvas e passadas para a *Activity* principal, que possui uma

View baseada na API do Google Maps, com o mesmo estilo de tela que pode ser observada na Figura 12.

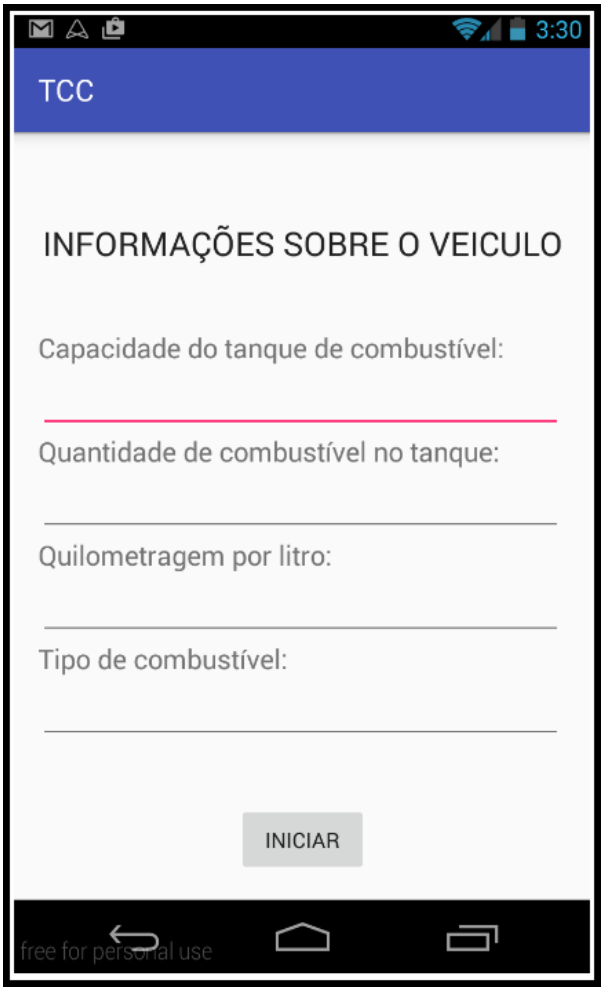

Figura 17 – Tela Inicial

Esta atividade reúne todas as funcionalidades das versões anteriores, porém adaptadas para ser possível a busca por uma rota que contenha os postos de combustível que estão situados nessa rota. Para que os postos sejam mostrados no mapa gerado pela API do Google Maps, foi necessário fazer diferentes *Place Searchs*, devido ao raio de busca ser limitado à apenas 50km.

Então, após estas buscas, é feita uma filtragem nos locais obtidos, para que sejam mostrados apenas os postos que estão próximos de algum ponto da rota. Com isso, o aplicativo calcula a quantidade de quilômetros que o carro do usuário pode percorrer, baseado nas informações passadas previamente, e mostra até que ponto da rota ele pode chegar antes de seu combustível acabar, como pode ser

visto na Figura 18, que corresponde a um exemplo de busca por rota entre Alegre-ES e Vitória-ES, com um carro que tem um tanque que armazena 70L, com 10L armazenados, que tem média de consumo de 9km/L e está abastecido com gasolina.

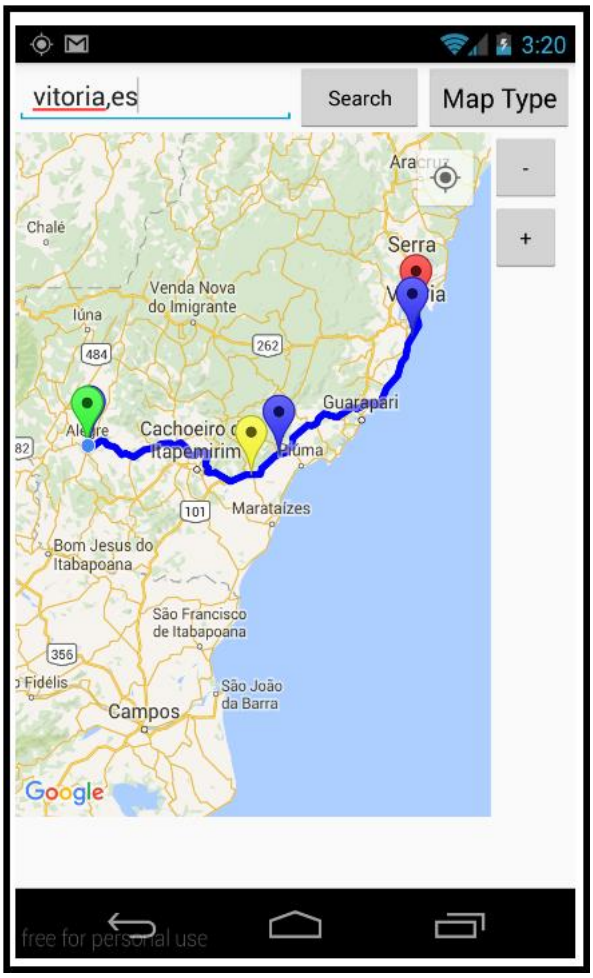

Figura 18 – Rota gerada

Na Figura 18, o ponto verde corresponde à saída, o vermelho à chegada, os azuis aos postos de combustível e o amarelo à distância máxima que o carro pode percorrer antes de precisar abastecer.

5. RESULTADOS OBTIDOS

Inicialmente foi proposta a criação de um aplicativo para a plataforma Android, que possibilitaria ao usuário criar um plano de abastecimento em uma viagem, através da utilização de informações retornadas pela API do Apontador (Apontador, 2014), que permitia a busca por um determinado local e retornava informações como nome, latitude e longitude e o preço dos seus produtos. Porém, após uma atualização na API, foram otimizadas algumas de suas funcionalidades e retirada a opção de retorno do preço dos produtos. Com isso, o aplicativo se tornou incapaz de criar um plano de abastecimento para o usuário, visto que não era mais possível saber o preço dos combustíveis fornecidos pelos postos ao longo da rota gerada, assim como não era viável, em questão de tempo e complexidade, a busca por uma nova implementação utilizando o banco de dados de postos de combustível fornecido pela Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP, 2015), em conjunto com os mapas fornecidos pela API do Google Maps.

Dessa forma, o aplicativo construído, nomeado como Postos na Viagem, permite ao usuário encontrar a menor rota entre a localização atual e um ponto de chegada, além de mostrar o ponto máximo da rota que o veículo conseguirá chegar com a quantidade de combustível atual e mostrará os postos de combustíveis ao longo da rota, como pode ser visto na Figura 18.

5.1. Testes e Validação

Para testar o funcionamento do aplicativo, foi utilizada uma maquina virtual chamada *Genymotion* (Genymotion, 2015) que simulou o funcionamento de um dispositivo Andorid em um notebook Lenovo com as seguintes características:

- Processador Intel(R) Core(TM) i5-3230M de 2.60GHz.
- Memória RAM de 4Gb.
- Sistema operacional Windows 8.1 de 64bits.
- Suporte a Toque Total Windows com 10 pontos de toque.
- HD de 1Tb.

O dispositivo simulado é um *smartphone* Google Nexus 4 com as seguintes características:

- Tela de 4,7” com resolução de 768x1280.
- Sistema Operacional Andorid 4.3.
- Processador Quad-Core de 1.5GHz.
- Memória RAM de 2Gb.
- Armazenamento de 8Gb.

Foram testadas 3 rotas diferentes a partir de 2 pontos de partida diferentes, e os resultados obtidos foram validados através de comparações com as mesmas rotas a partir dos mesmos pontos de partida gerados pelo Google Maps para *web*, assim, o aplicativo se mostrou capaz de gerar rotas ótimas praticamente idênticas às geradas pelo Google, como pode ser visto na Figura 19 que contém rotas com ponto inicial Alegre-ES, à esquerda estão os resultados do Google e à direita os resultados do aplicativo desenvolvido.

Na Figura 20, estão as comparações feitas com ponto inicial Vitoria-ES, à esquerda estão os resultados do Google e à direita os resultados do aplicativo desenvolvido.

Assim, a partir destes testes foi gerada a Tabela 1 com a comparação das distâncias e durações, estimadas, obtidas pelo Google Maps para *web* e o aplicativo Postos na Viagem, o que comprova que as rotas geradas são boas.

A busca por postos de combustível foi validada a partir de testes com buscas por postos de combustível no Google Maps para *web* e no aplicativo Preço de Combustíveis (2015). Assim, o aplicativo se mostrou capaz de achar os resultados superiores ao Google e ao aplicativo Preços de Combustíveis, visto que o mesmo não mostra alguns postos em localidades no interior do estado do Espirito Santo, como pode ser observado na Figura 21, resultados do Google à esquerda, resultados do Preço de Combustíveis ao centro e resultados do aplicativo desenvolvido à direita.

Então, a partir destes testes foi gerada a Tabela 2 com a comparação de números de postos obtidos pelo Google Maps para *web*, o aplicativo Preços de

Combustíveis e o aplicativo Postos na Viagem. Assim, comprovando que os resultados obtidos são satisfatórios.

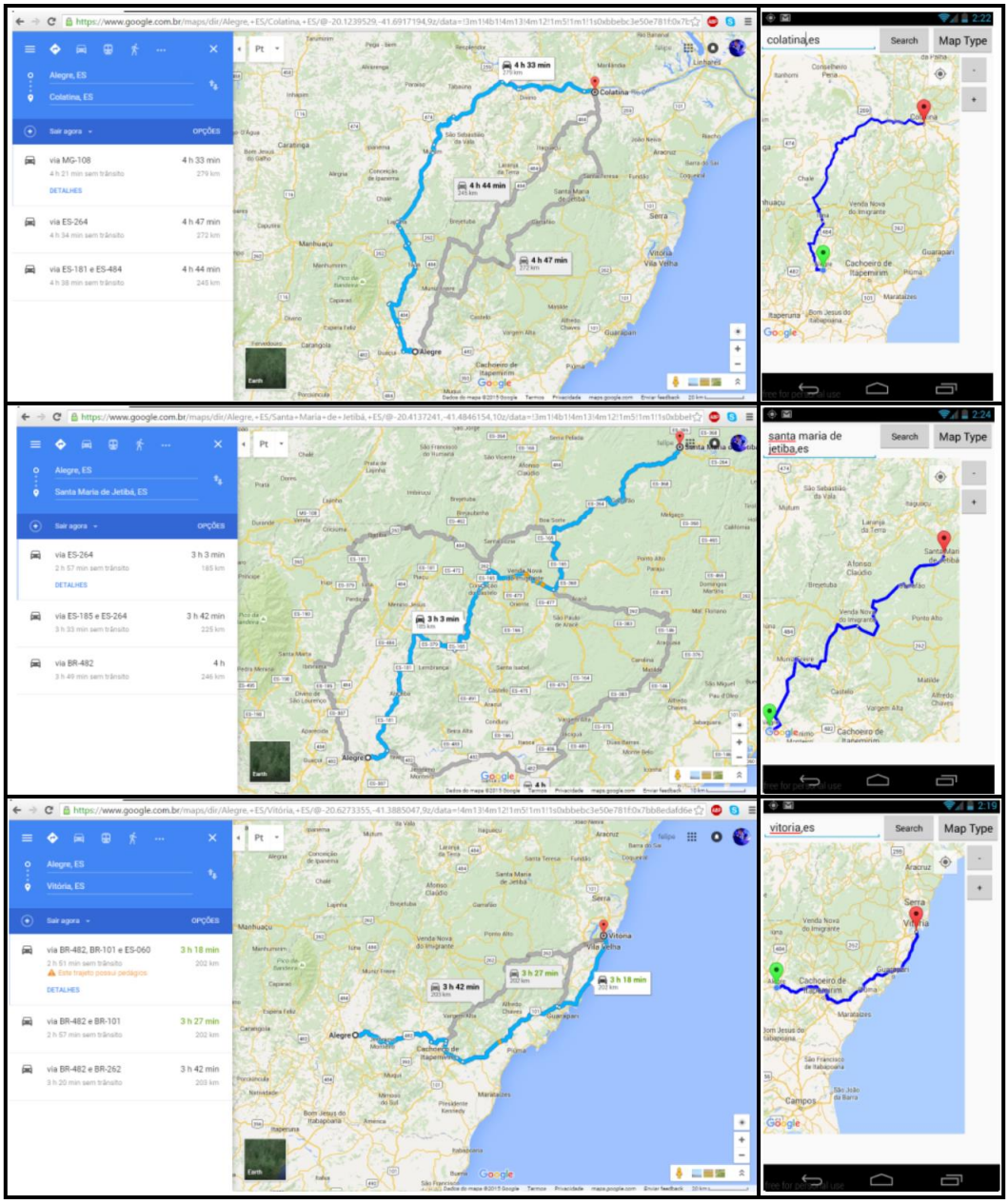

Figura 19 – Rotas a Partir de Alegre-ES.

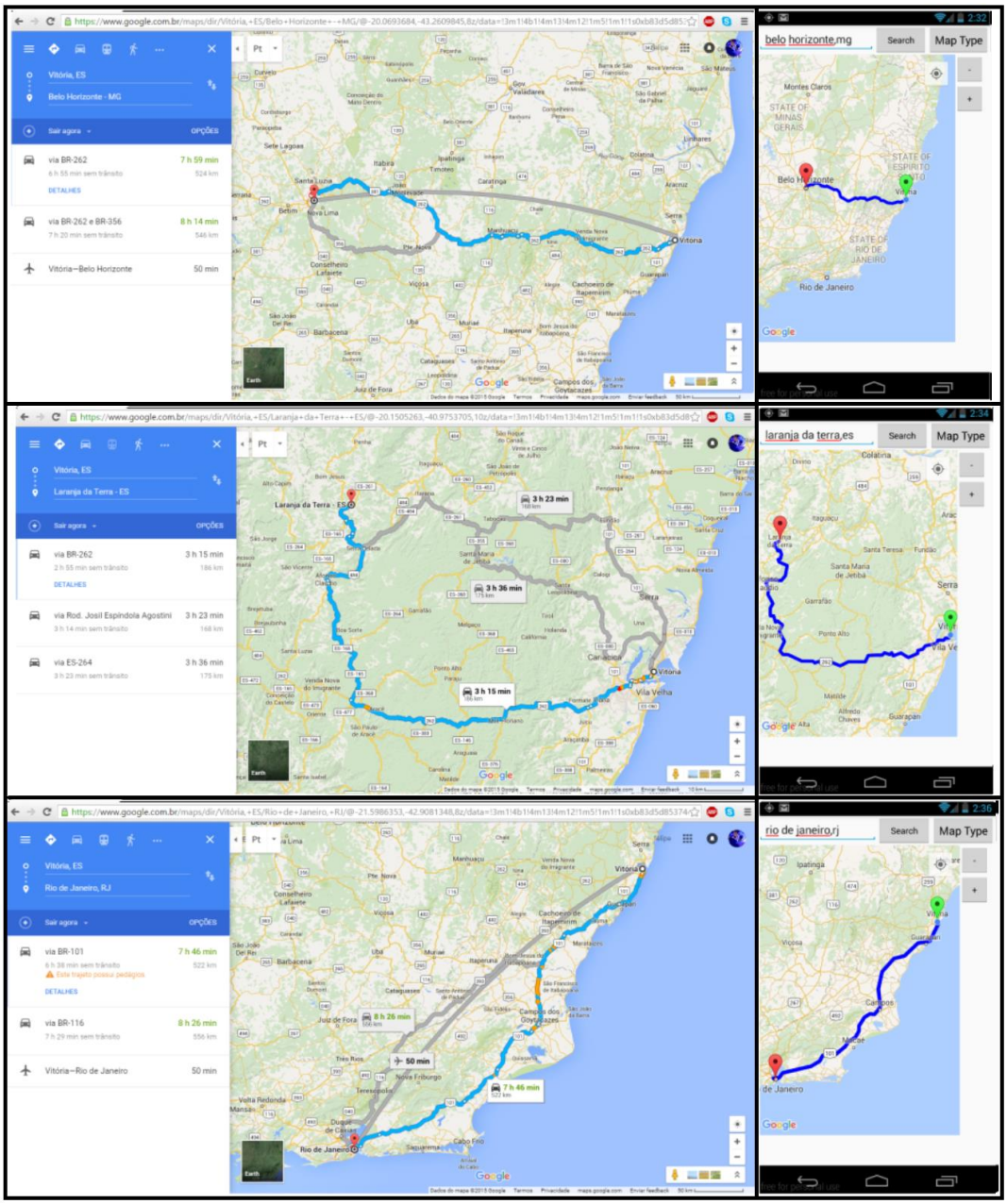

Figura 20, Rotas a Partir de Vitoria-ES.

Tabela 1 – Comparação Entre Resultados do Google Maps para *web* e o aplicativo Postos na Viagem

Rotas	Google Maps para *web*		App. Posotos na Viagem	
	Duração	Distância	Duração	Distância
Alegre-ES , Colatina-ES	4,51h	279km	4,49h	276,59km
Alegre-ES , Stª M. de Jetibá-ES	3,05h	185km	3,05h	185,9km
Alegre-ES , Vitoria-ES	3,25h	202km	3,24h	201,84km
Vitória-ES , Belo Horizonte-MG	7,88h	524km	7,87h	522,6km
Vitória-ES , Laranja da Terra-ES	3,18h	186km	3,17h	185,29km
Vitória-ES , Rio de Janeiro-RJ	7,2h	522km	7,5h	522,22km

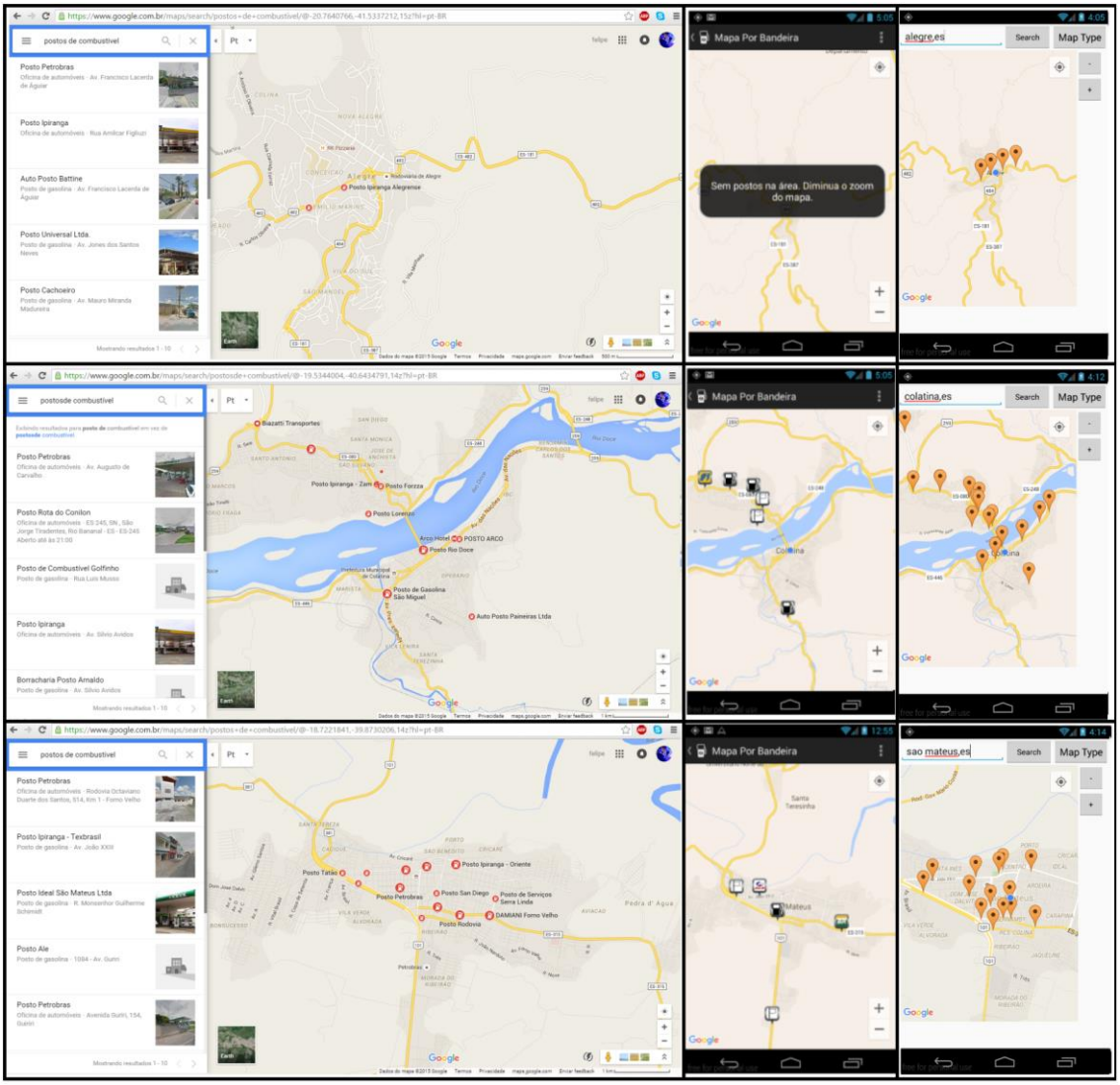

Figura 21- Comparação de Busca Por Postos de Combustível

Tabela 2 – Comparação do Número de Postos de Combustível Obtidos

Cidade da Busca	**Número de Postos Obtidos na Busca**		
	Google Maps para *web*	**App. Postos de Combustíveis**	**App. Postos na Viagem**
Alegre-ES	2 postos	0 postos	4 postos
Colatina-ES	9 postos	6 postos	16 postos
São Mateus-ES	13 postos	4 postos	14 postos

6. CONCLUSÃO

Com base no contexto do problema proposto, nos objetivos estabelecidos e nos testes feitos, o resultado alcançado se demonstra satisfatório, visto que o aplicativo apresenta uma forma de obter uma rota de menor distância entre dois pontos e informar a localização de postos de combustível ao longo da mesma, assim como o ponto máximo que se é possível chegar com uma determina quantidade de combustível em um veículo. Com isso ele pode se tornar uma boa ferramenta para ajudar usuários a planejar viagens.

Por fim, pode-se notar que é relevante a continuação do trabalho, no sentido de incrementar suas funcionalidades, ao se utilizar o banco de dados de postos de combustível da ANP, possibilitando assim a criação de um plano de abastecimento que realmente ajude os usuários a economizar recursos, além de melhorar a experiência entre o usuário e o aplicativo através da aplicação de conceitos de IHC (Interação Humano Computador). Pode-se adicionar também outras facilidades, como exemplo um cadastro prévio de modelos de carro, para assim melhorar ainda mais a experiência com o aplicativo.

O código fonte do aplicativo está disponível em: http://felipemk.webnode.com.br/news/tcc-aplicatio-movel-para-otimizacao-de-gastos-de-combustiveis/.

7. REFERÊNCIAS BIBLIOGRÁFICAS

ANDROID STUDIO. **Developer**, 2014. Disponível em: <https://developer.android.com/sdk/index.html> Acesso em: Junho de 2015.

ANGUISH, S.; BUCK, E.; YACKTMAN, D. **Cocoa Programming**. 1st. ed. [S.1.]: Sams Publishing, 2002.

ANP. **ANP**, 2015. Disponível em: <http://www.anp.gov.br/> Acesso em: Outubro de 2015.

APPBRAIN. **Number of Android applications**, 2014. Disponível em: <http://www.appbrain.com/stats/number-ofandroid-apps> Acesso em: Junho de 2015.

APONTADOR. **Apontador API**, 2014. Disponível em: <https://api.apontador.com.br/docs/> Acesso em: maio de 2015.

APONTADOR V2. **Apontador API Versão 2**, 2015. Disponível em: <https://api.apontador.com.br/v2/docs/> Acesso em: outubro de 2015.

BRAY, T., PAOLI, J., SPERBERG-MCQUEEN, C.M., MALER, E., YERGEAU, F., COWAN, J. **Extencible Markup Lenguage (XML)**. 2ª es. W3C. 2006. Recommendation.

BUYYA, R., YEO, C. S., VENUGOPAL, S., BROBERG, J., e BRANDIC, I. **Cloud computing and emerging it platforms**: Vision, hype, and reality for delivering computing as the 5th utility. Future Gener. Comput. Syst, 2009.

CNT. **Preços dos combustíveis mantém trajetória de alta**, 2015. Disponível em: <http://www.cnt.org.br/Paginas/Agencia_Noticia.aspx?noticia=precos-dos-combustiveis-mantem-trajetoria-de-alta-cnt> Acesso em: Dezembro de 2015.

DASH. **Dash**, 2015. Disponível em: <https://dash.by/> Acesso em: Novembro de 2015.

DEVELOPER, **Google Developer**, 2015. Disponível em: <https://developers.google.com/> Acesso em: Junho de 2015.

DUBSMASH. **Aplicativo Dubsmash**, 2015. Disponível em: <http://www.baixaki.com.br/android/download/dubsmash.htm> Acesso em: Junho de 2015.

ECONOMIA UOL. **Com alta de mpostos gasolina sobe R$0,22 por litro e diesel R$0,15**, 2015. Disponível em: <http://economia.uol.com.br/noticias/redacao/2015/01/19/com-alta-de-impostos-gasolina-sobe-r-022-por-litro-e-diesel-r-015.html> Acesso em: Maio de 2015.

FACEBOOK. **Aplicativo Facebook para Android**, 2014. Disponível em: <http://www.baixaki.com.br/android/download/facebook-para-android.htm> Acesso em: Junho de 2015.

FECOMBUSTIVEIS. **Motoristas buscam soluções para driblar aumento no preço da gasolina**, 2015. Disponível em: <http://www.fecombustiveis.org.br/clipping/motoristas-buscam-solucoes-para-driblar-aumento-no-preco-da-gasolina> Acesso em: Abril de 2015.

FLING, B. **Mobile Desin and Development**. Sebastopol: O'Reilly, 2009.

FREITAS, L.M. **Android** – Desenvolvendo aplicativos para dispositivos móveis. Hachi Tecnologia, 2012. Apostila.

G1. **Preço da gasolina no Brasil está quase 70% acima do internacional**, 2015. Disponível em: <http://g1.globo.com/economia/noticia/2015/01/preco-da-gasolina-no-brasil-esta-quase-70-acima-do-internacional.html> Acesso em: Abril de 2015.

GARGENTA, M. **Learning Android**. [S.1.]: O'Reilly Media, 2011.

GASNOL, **Aplicativo Gasnol**, 2015. Disponível em: <http://www.techtudo.com.br/tudo-sobre/gasnol.html> Acesso em: Dezembro de 2015.

GAZETA DO POVO. **Passadas as eleições Brasil caminha para a maior carga tributária da historia**, 2015. Disponível em: <http://www.gazetadopovo.com.br/vida-publica/passadas-as-eleicoes-brasil-caminha-para-a-maior-carga-tributaria-da-historia-ehdajgmtepm1tjntm8w32awy6> Acesso em: Maio de 2015.

GENYMOTION. **Genymotion**, 2015. Disponível em: <https://www.genymotion.com/#!/> Acesso em: Novembro de 2015.

GERSTING, J. L. **Fundamentos Matemátios para a Ciência da Computação**, Tradução da 2ª Edição, 2014.

GOOGLE. **Google Maps API**, 2015. Disponível em: <https://developers.google.com/maps/?hl=pt-br> Acesso: Maio de 2015

GOOGLE DEVELOPER. **Google Maps API Documentation**, 2015. Disponível em: <https://developers.google.com/maps/documentation/javascript/directions?hl=pt-br#Directions> Acesso: Maio de 2015.

GOOGLE DIRECTIONS. **Google Directions exemple**, 2015. Disponível em: <https://developers.google.com/maps/documentation/javascript/examples/directions-simple> Acesso: Novembro de 2015.

GOOGLE DIRECTIONS E ELEVATION API. **Google Directions e Elevation API –** Aperfeiçoando o cálculo de autonomia de veículos, 2010. Disponível em: <http://imasters.com.br/artigo/17349/apis-google/google-directions-e-elevation-api-aperfeicoando-o-calculo-de-autonomia-de-veiculos/> Acesso: Setembro de 2015.

GOOGLE MAPS API. **Google Maps Android API**, 2015. Disponível em: <https://developers.google.com/maps/documentation/android-api/?hl=pt-br> Acesso: Setembro de 2015.

GOOGLE PLACES. **Place Search**, 2015. Disponível em: <https://www.google.com.br/maps/place/Posto+Moby+Dick/@-20.3368335,-40.3483709,13z/data=!4m5!1m2!2m1!1sposto+de+combustivel!3m1!1s0x0000000000000000:0xceb7a6be6af39135> Acesso: Setembro de 2015.

IDC. **Press Release**. Disponível em: < http://www.idc.com/getdoc.jsp?containerId=prUS24676414> Acesso em: Abril de 2015.

INSTAGRAM. **Aplicativo Instagram**, 2014. Disponível em: <http://www.baixaki.com.br/android/download/instagram.htm> Acesso em: Junho de 2015.

JUNG, C. F. **Metodologia Aplicada a Projetos de Pesquisa**: Sistemas de Informação & Ciência da Computação. Rio de Janeiro – RJ: Axcel Books do Brasil Editora, 2004.

LECHETA, R. R. **Google Android** - aprenda a criar aplicações para dispositivos móveis com o Android SDK. 3.ed. São Paulo: Ed. Novatec, 2013.

JOGOS. **Jogos para Android**, 2015. Disponível em: <http://www.baixaki.com.br/android/jogos/ > Acesso em: Junho de 2015.

MACIEL, M. S. **Tributos Incidentes sobre combustíveis**, 2011.

MALAQUIAS, N. G. L. **Uso dos Algoritmos Genéticos para a Otimização de Rotas de Distribuição**, 2006.

MILLWARD BROWN. **adReaction**, 2014. Disponível em: <http://www.millwardbrown.com/adreaction/2014> Acesso em: Novembro de 2015.

MINASPETRO. **Em novembro, preço do combustível subiu 7,96%.** Gasolina teve maior alta no Nordeste e o álcool no Sudeste, 2015. Disponível em: <http://minaspetro.com.br/noticia/em-novembro-preco-do-combustivel-subiu-796-gasolina-teve-maior-alta-no-nordeste-e-o-alcool-no-sudeste/> Acesso em: Dezembro de 2015.

MORESI, E. **Metodologia de Pesquisa**. Brasília, DF: Universidade Católica de Brasília – Pró-reitora de Pós-graduação – Programa de Pós-graduação Stricto Sensu em Gestão do Conhecimento e Tecnologia da Informação, 2003.

NOTÍCIAS AUTOMOTIVAS, **Gasolina**: Combustível tem nova alta de preços, 2015. Disponível em: <http://www.noticiasautomotivas.com.br/gasolina-combustivel-tem-nova-alta-de-precos> Acesso em: Março de 2015.

NETO, P. O. B. **Teoria e Modelo de Grafos**. 4ª ed. Edgard Blucher, 2006.

NOTÍCIAS R7. **Economistas esperam reajuste de até 95% na gasolina neste ano**, 2015. Disponível em: <http://noticias.r7.com/economia/economistas-esperam-reajuste-de-ate-95-na-gasolina-neste-ano-24052015> Acesso em: Abril de 2015.

ORACLE. **Java Timeline**. 2012. Disponível em: <http://oracle.com.edgesuite.net/timeline/java/> Acesso em: Junho de 2015.

PALCOM MP3. **Aplicativo Palco MP3**, 2014. Disponível em: <http://www.baixaki.com.br/android/download/palco-mp3.htm> Acesso em: Junho de 2015

PENSECARROS. **Sete passos para economiar combustível na hora de viajar**, 2014. Disponível em: <http://revista.pensecarros.com.br/noticia/2014/04/confira-sete-passos-para-economizar-combustivel-na-hora-de-viajar-4477373.html> Acesso em: Maio de 2015.

PEREIRA, L. C. O., & SILVA, M. L. **Andorid Para Desenvolvedores**. Rio de Janeiro: Braspost, 2009.

PETROBRÁS. **A Gasolina Brasileira**, 2015. Disponível em: <http://www.br.com.br/wps/wcm/connect/portal+de+conteudo/produtos/automotivos/gasolina/como+esta+o+preco+da+gasolina+brasileira+em+relacao+a+outros+paises+da+america+latina+e+do+mundo> Acesso em: Maio de 2015.

PINHEIRO, D. H. **Desenvolvimento de um Modelo de Otimização de Consumo de Combustível para o Problema de Roteamento de Veículos Capacitados**, 2013.

PREÇO DE COMBUSTÍVEIS. **Aplicativo Preço de Combustíveis**. 2015. Disponível em: <https://play.google.com/store/apps/details?id=br.com.precodoscombustiveis.pdc&hl=pt_BR> Acesso em: Novembro de 2015.

RICMAIS. **Depois de aumento motoristas são forçados a buscar alternativas que visam economizar combustível**, 2015. Disponível em: <http://ricmais.com.br/sc/economia/videos/depois-do-aumento-motoristas-sao-forcados-a-buscar-alternativas-que-visam-economizar-combustivel> Acesso em: Maio de 2015.

RUNTASTIC. **Runtastic**, 2015. Disponível em: <https://www.runtastic.com> Acesso em: Novembro de 2015.

SILVA, E. L., & MENEZES, E. M. **Metodologia da Pesquisa e Elaboração de Dissertação**. 3ª ed. Florianópolis: Laboratório de Ensino a Distância da UFSC, 2001.

SILVA, L.A. **Apostila de Android** – Programando passo a passo. Rio de Janeiro, 2010. Apostila.

SINESP. **Aplicativo Sinesp**, 2014. Disponível em: <http://www.baixaki.com.br/android/download/sinesp-cidadao.htm> Acesso em: Junho de 2015.

TECHTUDO. **Android domina mercado Brasileiro**, 2012. Disponível em: <http://www.techtudo.com.br/noticias/noticia/2012/09/android-domina-mercado-brasileiro-windows-phone-mais-expressivo-que-iphone.html> Acesso em: Abril de 2015.

UOL. **É possível economizar até 33% de combustível, saiba como**, 2014. Disponível em: <http://noticias.carsale.uol.com.br/ultimas/2014/04/19/e-possivel-economizar-ate-33-de-combustivel-saiba-como> Acesso em: Abril de 2015.

W3C. **Web Servises Architecture**, 2004. Disponível em: <http://www.w3.org/TR/ws-arch/> Acesso: Otubro de 2015.

WHATSAPP, **Aplicativo Whatsapp**, 2014. Disponível em: <https://www.whatsapp.com/?l=pt_br> Acesso em: Junho de 2015.

WOSLEY, L. A. **Integer programming**, 1 Ed. Wiley Interscience, 1998. ISNB: 9780471283669.

WEB SERVICE, **Web Service – API**, 2014. Disponível em: <http://wiki.teqa.nl/display/V4/Webservice+-+API> Acesso em: novembro de 2015.

XML, **XML Syntax**, 2015. Disponível em: <http://www.w3schools.com/xml/xml_syntax.asp> Acesso: Novembro de 2015.

ZIVIANI, N. **Projeto de Algoritmos** – com implementação em Pascal e C. 2.ed. São Paulo: Ed. Thomson, 2004.