

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO

**DESENVOLVIMENTO DE UMA SOLUÇÃO ANDROID DE
ORIENTAÇÃO GEOGRÁFICA PARA REPRESENTANTES
COMERCIAIS COM A UTILIZAÇÃO DE GPS E GOOGLE
MAPS**

LEANDRO IVAN CLEBSCH

BLUMENAU
2014

2014/2-09

LEANDRO IVAN CLEBSCH

**DESENVOLVIMENTO DE UMA SOLUÇÃO ANDROID DE
ORIENTAÇÃO GEOGRÁFICA PARA REPRESENTANTES
COMERCIAIS COM A UTILIZAÇÃO DE GPS E GOOGLE
MAPS**

Trabalho de Conclusão de Curso submetido à
Universidade Regional de Blumenau para a
obtenção dos créditos na disciplina Trabalho
de Conclusão de Curso II do curso de Sistemas
de Informação— Bacharelado.

Prof. Jhony Alceu Pereira, Especialista - Orientador

**BLUMENAU
2014**

2014/2-09

**DESENVOLVIMENTO DE UMA SOLUÇÃO ANDROID DE
ORIENTAÇÃO GEOGRÁFICA PARA REPRESENTANTES
COMERCIAIS COM A UTILIZAÇÃO DE GPS E GOOGLE
MAPS**

Por

LEANDRO IVAN CLEBSCH

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente: _____
Prof. Jhony Alceu Pereira, Especialista – Orientador, FURB

Membro: _____
Prof. Cláudio Ratke, Mestre – FURB

Membro: _____
Prof. Miguel Alexandre Wisintainer, Mestre – FURB

Blumenau, 4 de dezembro de 2014.

Dedico este trabalho a todos os amigos,
especialmente aqueles que me ajudaram
diretamente na realização deste.

AGRADECIMENTOS

A Deus, pelo seu imenso amor e graça, que me concedeu a capacidade para realizar este trabalho.

À minha família, pelo apoio e incentivo.

Ao meu orientador, professor Jhony, por ter acreditado na conclusão deste trabalho sempre fazendo com que as coisas parecessem mais simples do que pareciam.

Ao professor Wilson, pelas dicas para a realização deste trabalho.

Aos professores do Departamento de Sistemas e Computação da Universidade Regional de Blumenau por suas contribuições durante os semestres letivos.

Embora ninguém possa voltar atrás e fazer um novo começo, qualquer um pode começar agora a fazer um novo fim!

Chico Xavier

RESUMO

Um bom aplicativo para representantes comerciais deve levar em consideração, além da simplicidade, um conteúdo que lhe auxilie no desenvolvimento de suas atividades. Dispositivos com cada vez mais recursos embarcados, os aplicativos devem levar em consideração utilizar todo o potencial do dispositivo, facilitando as tarefas do usuário. Este trabalho promove uma proposta de aplicativo móvel para orientação geográfica, com a utilização de Google Maps para representantes comerciais, desenvolvido na plataforma Android, utilizando o recurso de geolocalização por *Global Position System* (GPS) e integrado as *Application Programming Interface* (API's) do Google Maps. Com estes recursos, o aplicativo irá auxiliar os representantes na indicação de rotas, otimizando o seu tempo com deslocamento entre os clientes e facilitando o planejamento de visitas. Dentre outras funcionalidades, o aplicativo contém o cálculo de tempo entre os deslocamentos. Este recurso facilitará ao representante a organizar-se, a otimizar seu tempo e ajudar na economia de recursos.

Palavras-chave: Android. Aplicações móveis. Representante comercial.

ABSTRACT

A good application for trade representatives must take into account beyond the simplicity, content that helps you in the development of its activities. Devices with increasingly filled features, applications should consider using any device potential facilitating the user's life. This work promotes a mobile application proposal for trade representatives, developed on the Android platform using the geolocation feature integrated Global Position System (GPS) and Google Maps Application Programming Interface (API). With these resources, the application will assist representatives in the indication of routes, optimizing your time with displacement between customers and facilitate planning visits. Among other features the application contains the calculation time between shifts. Feature that will facilitate the representative to organize, optimize time and help in saving resources.

Key-words: Android. Mobile applications. Trade Representative.

LISTA DE FIGURAS

Figura 1 - Informações do pedido.....	22
Figura 2 - Consulta de linhas de ônibus	23
Figura 3 - Relatórios e gráficos Base CRM.....	24
Figura 4 - Cliente exibido no mapa	24
Figura 5 - Funcionamento do sistema	26
Figura 6 - Diagrama de casos de uso.....	28
Figura 7 - Diagrama de atividades.....	29
Figura 8 - Diagrama de classes.....	30
Figura 9 - IDE Eclipse com <i>Android Developer Tools</i>	32
Figura 10 - Inicializa mapa.....	33
Figura 11 - Menu principal.....	34
Figura 12 - Formulário de cadastro de cliente	35
Figura 13 - Código para converter endereço em coordenadas	35
Figura 14 - Coordenadas convertidas	36
Figura 15 - Cadastro realizado	36
Figura 16 - Lista de clientes	37
Figura 17 - Método que gera a lista de clientes no visor.....	37
Figura 18 - Menu interação da tela de clientes.....	38
Figura 19 - Mensagem de confirmação da atualização	38
Figura 20 - Confirmação de exclusão.....	39
Figura 21 - Mensagem de confirmação da exclusão	39
Figura 22 - Confirmação de agendamento	39
Figura 23 - Mensagem de cliente agendado	40
Figura 24 - Método que copia cliente para a tabela da agenda.....	40
Figura 25 - Tela da agenda	41
Figura 26 - Menu do agendamento selecionado.....	41
Figura 27 - Confirmação de exclusão.....	42
Figura 28 - Mensagem de confirmação da exclusão	42
Figura 29 - Cadastro de ponto de interesse.....	43
Figura 30 - Método que cadastra ponto de interesse	43
Figura 31 - Coordenada do ponto de interesse	44

Figura 32 - Confirmação cadastro ponto de interesse	44
Figura 33 - Melhor rota	45
Figura 34 - Método inicializa do mapa.....	46
Figura 35 - Método que carrega clientes agendados no mapa.....	46
Figura 36 - Método para traçar a rota	47

LISTA DE QUADROS

Quadro 1 - Tecnologia dos principais sistemas operacionais para dispositivos móveis	19
Quadro 2 - Requisitos Funcionais	26
Quadro 3 - Requisitos não funcionais	27
Quadro 4 - Comparativo entre aplicações	48
Quadro 5 - Caso de uso Manter clientes.....	55
Quadro 6 - Caso de uso Manter agenda.....	56
Quadro 7 - Caso de uso Manter ponto de interesse	57
Quadro 8 - Caso de uso Cálculo de melhor rota.....	57

LISTA DE SIGLAS

AAC – *Advanced Audio Coding*

ADT – *Android Development Tools*

API – *Application Programming Interface*

APP – *Mobile Application*

CRM - *Customer Relationship Management* (Gestão de Relacionamento com o Cliente)

GPS – *Global Position System*

IDE – *Integrated Development Environment*

JVM – *Java Virtual Machine*

KML – *Keyhole Markup Language*

MMS – *Multimedia Messaging Service*

MP3 – *Moving Picture Audio Layer 3*

MPEG – *Moving Picture Experts Group*

MVC – *Model View Controller*

PHP – *Personal Home Page*

POI – *Point Of Interest*

RF – *Requisito Funcional*

RN – *Regra de Negócio*

RNF – *Requisito Não Funcional*

SDK – *Software Development Kit*

SMS – *Short Message Service*

SQL – *Structured Query Language*

UC – *Use Case*

UML – *Unified Modeling Language*

VGA – *Video Graphics Array*

XML - *eXtensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO.....	12
1.1 OBJETIVOS DO TRABALHO	13
1.2 ESTRUTURA DO TRABALHO	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1. GPS.....	15
2.2. GOOGLE MAPS.....	16
2.3. ANDROID.....	17
2.4. APLICAÇÕES MÓVEIS	18
2.5. REPRESENTANTES COMERCIAIS	20
2.6. TRABALHOS CORRELATOS	21
3 DESENVOLVIMENTO DA FERRAMENTA	25
3.1 LEVANTAMENTO DE INFORMAÇÕES	25
3.1.1 Requisitos principais	26
3.2 ESPECIFICAÇÃO	27
3.2.1 Diagrama de casos de uso	27
3.2.2 Diagrama de atividades	28
3.2.3 Diagrama de classes	29
3.3 IMPLEMENTAÇÃO	30
3.3.1 Técnicas e ferramentas utilizadas.....	31
3.3.1.1 Ferramenta de desenvolvimento Android.....	31
3.3.1.2 API Google Maps versão 2.....	32
3.3.2 Operacionalidade da implementação	33
3.4 RESULTADOS E DISCUSSÃO	47
4 CONCLUSÕES.....	50
4.1 EXTENSÕES	51
REFERÊNCIAS	52
APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO	55

1 INTRODUÇÃO

Os mapas são a mais antiga representação do pensamento geográfico sendo registrados desde o início das primeiras sociedades e registrados de formas diferentes das que conhecemos atualmente. Entretanto, os objetivos continuam basicamente os mesmos: facilitar a localização, orientação e movimentação. De acordo com Almeida, Guerrero e Fiori (2007, p. 21), o surgimento do mapa precede a invenção da escrita. Estes eram confeccionados para mostrar as áreas onde haviam caça e pesca ou marcar o caminho para casa.

O mapa é uma imagem convencionalizada, codificada que representa feições e características da realidade geográfica. O mapa não reproduz fielmente o terreno. Ele é a construção que seleciona alguns aspectos e os representa, fazendo uso de um sistema de símbolos. Os elementos espaciais da superfície terrestre escolhidos para serem representados no mapa (como objetos, fatos e relações) são transformados em símbolos e localizados no mapa a partir de um sistema de coordenadas que considera distâncias e direções (ALMEIDA; GUERRERO; FIORI, 2007, p. 20).

Tem-se como alguns dos principais exemplos de aplicações do geoposicionamento em transporte o planejamento da capacidade e manutenção de vias; posicionamento para colheita de grãos; acompanhamentos estatísticos de transporte público; criação de rotas para cargas intermunicipais e interestaduais e; acompanhamento por *Global Position System* - Sistema de Posicionamento Global (GPS), para citar algumas (CARVALHO, 2002). Dentro destes, o GPS vem se popularizando nos últimos tempos, sendo um sistema capaz de fornecer informações sobre a localização de pontos utilizando-se de satélites posicionados em órbita na Terra. O mesmo pode fornecer as coordenadas aproximadas da localização de uma pessoa, por exemplo, que se encontra em qualquer parte do mundo (DANA, 1999).

A história deste sistema se iniciou em 1957, ano em que a União Soviética lançou o primeiro satélite artificial da história. Fato que deu início aos primeiros estudos sobre o uso de satélites na localização de pontos sobre a superfície terrestre. Contudo, foram os americanos que, de fato, criaram o sistema. A base dessa criação foi o projeto NAVSTAR, desenvolvido em 1960 pelo Departamento de Defesa dos Estados Unidos. O sistema oferecia diversas informações sobre qualquer parte do planeta – como localização e clima – a qualquer hora do dia, algo que era de grande importância para o uso militar. Após vários ajustes e correções, o projeto NAVSTAR veio se tornar totalmente operacional apenas em 1995 (HISTÓRIA DE TUDO, 2013).

Com esta tecnologia, a empresa estadunidense Google desenvolveu uma ferramenta, utilizando este recurso, chamada Google Maps. O Google Maps é uma ferramenta que possibilita visualizar mapas e fotos de satélite de qualquer lugar do planeta em computadores e dispositivos móveis. Esta ferramenta é disponibilizada de forma gratuita ao público em geral (GUIMARÃES, 2013).

Representantes comerciais podem atuar, com grande sucesso, em quase todas as áreas municipais e também áreas de atuação profissional tais como, serviço, intercâmbio, biblioteca, fábricas, oficinas, entidades religiosas, zona rural, clubes, jornais e revistas. Em todas estas áreas tem-se a necessidade de deslocamento e localização. (INSTITUTO NACIONAL DO LIVRO, 1972).

Muitas vezes deslocar-se de um cliente a outro, significa percorrer quilômetros de distância e, com isto, dispender boa parte do tempo para a localização do cliente tendo um gasto elevado no consumo de combustível. Devido a estes fatos, muitas vezes o representante não consegue ter a visão da melhor rota entre os clientes. Sendo assim, estes profissionais se deparam com a dificuldade para formularem sua rota pessoal de visitas aos clientes de maneira a maximizar seu tempo e diminuir seus custos operacionais. Outro problema encontrado por representantes é que, dos sistemas encontrados, poucos se enquadram na perspectiva de baixo custo de investimento para o pequeno ou autônomo profissional. E, destes, a dificuldade de tê-los em nível de acesso personalizado e plataforma mobile, que possa acompanhar o representante em suas visitas.

Diante dos fatos apresentados e, aliado ao fato de poucos sistemas oferecem uma solução de localização geográfica especificamente para representantes comerciais, será desenvolvida uma aplicação móvel para melhorar a organização, integrada ao Google Maps.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é o desenvolvimento de uma aplicação móvel na plataforma Android, de orientação geográfica, para representantes comerciais com a utilização de GPS e Google Maps. O sistema permitirá programar as visitas de acordo com os clientes agendados, representando-os em pontos no mapa com a rota a ser percorrida entre eles, fornecida pela API do Google Maps. Para auxiliar na programação da agenda, será exibido o tempo estimado para o deslocamento e o cadastro de pontos de interesse próximos ao cliente. Desta forma

auxiliar os representantes a diminuir o tempo entre os atendimentos e diminuir os gastos com combustível.

1.2 ESTRUTURA DO TRABALHO

No primeiro capítulo tem-se a introdução ao tema principal deste trabalho com a apresentação da justificativa e dos objetivos.

No segundo capítulo apresenta-se a fundamentação teórica pesquisada sobre GPS, Google Maps, plataforma Android, aplicações móveis, representantes comerciais e trabalhos correlatos.

O terceiro capítulo apresenta o desenvolvimento da ferramenta iniciando-se com o levantamento de informações com a especificação dos requisitos principais do sistema, tendo na sequência a especificação através dos diagramas. Em seguida é descrita a implementação com as técnicas e ferramentas usadas e a operacionalidade da implementação.

No quarto capítulo tem-se as conclusões deste trabalho bem como apresentam-se sugestões para trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Para entendimento, neste capítulo serão detalhados aspectos e conceitos da tecnologia GPS, ferramenta Google Maps, plataforma Android, aplicações móveis e representantes comerciais. Em seguida serão apresentados os trabalhos correlatos, bem como uma ferramenta existente no mercado com funções similares as da ferramenta desenvolvida.

2.1. GPS

O Sistema de Posicionamento Global (GPS) é um recurso baseado em sistema de navegação por satélite. Este sistema fornece a localização e informações de tempo, em qualquer lugar, onde há uma linha de visão desobstruída para quatro ou mais satélites GPS. O sistema fornece características específicas para usuários militares, civis e comerciais em todo o mundo. Ele é mantido pelo governo dos Estados Unidos e é de livre acesso a qualquer pessoa com um receptor GPS (ALEXANDROW, 2013).

A modernização do GPS tornou-se uma iniciativa para atualizar o Sistema de Posicionamento Global com novas capacidades para atender às crescentes necessidades militares, civis e comerciais. O sistema está sendo implementado através de uma série de aquisições, incluindo satélites GPS e a próxima geração do sistema de Controle Operacional (*Next Generation GPS Operation Control System - OCX*). O Governo dos Estados Unidos continua melhorando os equipamentos GPS dos recursos no espaço e segmentos na terra para aumentar o desempenho e precisão (AIR FORCE, 2011).

De acordo com David (1999), o GPS atual consiste em três segmentos principais. O segmento espacial, um segmento de controle e um segmento de usuários. A Força Aérea dos Estados Unidos desenvolve, mantém e opera os segmentos espaciais e controle. Satélites GPS transmitem sinais do espaço, e cada receptor GPS utiliza esses sinais para calcular a sua localização tridimensional (latitude, longitude e altitude).

O GPS foi um verdadeiro sucesso, fato que fez com que os Estados Unidos tenham disponibilizado as informações, antes somente de uso militar, para o uso civil e gratuito (HISTORIA DE TUDO, 2013).

Conforme Dana (1999), o segmento espacial é composto de 24 a 32 satélites em órbita

média, e incluindo também os adaptadores de carga para os reforços necessários para lançá-los em órbita. O segmento de controle é composto por uma estação de controle mestre, uma estação de controle alternativo mestre, e uma série de antenas terrestres dedicadas e compartilhadas e estações de monitoração. O segmento de usuário é composto de centenas de milhares de norte-americanos e usuários militares do seguro serviço de posicionamento GPS de precisão e dezenas de milhões de usuários civil, comercial, científica e do Serviço Padrão de Posicionamento.

2.2. GOOGLE MAPS

De acordo com Google (2013b), a aplicação destinada para localização e exibição de mapas, denominada Google Maps, é um serviço de pesquisa e visualização de mapas e imagens de satélite da Terra gratuito na web fornecido e desenvolvido pela empresa estadunidense Google.

A Google é uma empresa multinacional de serviços *online* e software dos Estados Unidos, fundada no começo de 1995 por Larry Page e Sergey Brin. Tudo começou quando estudavam Engenharia de Sistemas Informáticos em Stanford, e começam a trabalhar em conjunto num motor de pesquisa denominado BackRub. A Google hospeda e desenvolve uma série de serviços e produtos baseados na internet gerando lucro principalmente através da publicidade pelo AdWords (GOOGLE, 2013a).

De acordo com Google (2013a), em abril de 2002 foi lançado um conjunto de API's, que permite que os programadores consultem mais de dois milhões de documentos Web e programem no seu ambiente favorito, incluindo Android, Java, Perl e Visual Studio. Em março de 2004 foi apresentado o Google Local, que oferece fichas de empresas, mapas e direções relevantes nas proximidades. Hoje o Google Local é integrado no Google Maps. Atualmente, o serviço disponibiliza mapas e rotas para qualquer ponto nos Estados Unidos, Canadá, na União Europeia, Austrália e Brasil, entre outros. Disponibiliza também imagens de satélite do mundo todo, com possibilidade de um zoom nas grandes cidades, como Nova Iorque, Paris, São Paulo, Rio de Janeiro, Brasília, entre outras. Com uma conta Google, já é possível destacar as suas próprias rotas, pontos e áreas, gerar comentários e compartilhar os respectivos links de acesso ao mapa criado. Também é possível gerar um arquivo *Keyhole Markup Language* - KML para integração com o Google Earth.

Ainda conforme Google (2013b), a ferramenta foi desenvolvida por pensarem que os mapas seriam úteis e divertidos. Por isso foi planejado para simplificar a forma de ir do ponto A ao ponto B. Com o Google Maps é possível ver, por exemplo, hotéis mais próximos sendo visíveis de forma direta e fácil. Pode-se arrastar o mapa para ver a área adjacente de forma dinâmica e sem a espera para o carregamento de uma nova imagem.

Google em uma postagem em seu *blog* oficial afirma que, o Google Maps oferece a instrução passo-a-passo para onde você está indo. Se um determinado cruzamento na rota parece complicado, basta clicar nos botões para aproximar e ter uma visualização ampliada do local (GOOGLE, 2013b).

2.3. ANDROID

Para abranger o mercado a Google desejava lançar um aparelho com serviços baseados em localização, mas não possuíam plataforma para isso. Então em agosto de 2005 adquiriu a Android Inc, uma pequena empresa em Palo Alto (Califórnia – Estados Unidos) que desenvolvia uma plataforma para celulares baseado em Linux, com o objetivo de ser uma plataforma flexível, aberta e de fácil migração para os fabricantes. Com a união das tecnologias de GPS, Google Maps e Android, hoje temos acesso a uma ferramenta repleta de funcionalidades para auxiliar pessoas e empresas de todo o mundo. (GUIMARÃES, 2013).

Android é um sistema operacional baseado no núcleo do Linux desenvolvido para dispositivos móveis. Esta plataforma foi desenvolvida pela Open Handset Alliance, então liderada pelo Google e outras empresas. Segundo a Google, mais de 1 milhão e 300 mil aparelhos com este sistema operacional são ativados todos os dias. Utilizado por vários fabricantes de celulares como: HTC, Samsung, Sony, Motorola, LG e recentemente a Positivo Informática (LECHETA, 2010).

De acordo com LECHETA (2010), com o lançamento do SDK, características e especificações para o Android são facilmente distribuídas:

- a) *handset layouts* – a plataforma é adaptada tanto para dispositivos *Video Graphics Array* (VGA), maiores, gráficos 2D, bibliotecas gráficas 3D baseadas em OpenGL ES especificação 2.0 e os layouts mais tradicionais de smartphones;
- b) armazenamento – é utilizado SQLite para armazenamento de dados;
- c) mensagens – tanto SMS como MMS são formas disponíveis de envio de

- mensagens;
- d) navegador – o navegador disponível no sistema é baseado no *framework* de Código aberto conhecido como WebKit;
 - e) máquina virtual dalvik – aplicações escritas em Java são compiladas em *bytecodes* Dalvik e executadas usando a Máquina Virtual Dalvik, especializada e desenvolvida para uso em dispositivos móveis, o que permite que programas sejam distribuídos em formato binário (*bytecode*) e possam ser executados em qualquer dispositivo Android, independentemente do processador utilizado. Apesar das aplicações Android serem escritas na linguagem Java, ela não é uma máquina virtual Java, já que não executa *bytecode* JVM;
 - f) multimídia – o sistema suporta formatos de áudio e vídeo como: MPEG-4, H.264, MP3, e AAC;
 - g) suporte adicional de hardware – o Android é totalmente capaz de fazer uso de câmeras de vídeo, tela sensível ao toque, GPS, acelerômetros, e aceleração de gráficos 3D;
 - h) ambiente de desenvolvimento (SDK) – inclui um emulador, ferramentas para *debugging*, memória e análise de desempenho. O Eclipse (IDE) poderá ser utilizado através do *plugin Android Development Tools (ADT)*.

Com estes recursos é possível utilizar aplicativos que fazem uso das ferramentas embarcadas nos aparelhos celulares.

2.4. APLICAÇÕES MÓVEIS

Aplicações móveis são desenvolvidas e instaladas em dispositivos móveis tais como, telefone celular, *smartphone*, tablete ou um leitor de músicas, e é conhecido pela sigla “app” que, do inglês, significa *Application Software*. Estes dispositivos nos permitem o acesso à informação a qualquer momento e em qualquer lugar. Utilizada em vários domínios de aplicação por um usuário (que pode estar em movimento) e tem o objetivo de facilitar o dia-a-dia dos utilizadores. Com simples acesso às informações, proporciona uma quantidade infinita de possibilidades (AUGUSTIN, 2001).

De acordo com Criarenet (2012), as plataformas de aplicações móveis, a exemplo do

Android, rapidamente tornaram-se as mais populares para *smartphones* e dispositivos móveis. Cada vez mais empresas estão adotando a mobilidade como estratégia para seus negócios.

Nos dias atuais, existem diversos sistemas operacionais para dispositivos móveis, com diferentes fatias de mercado e diferentes métodos de desenvolvimento. O mais comum, de acordo com Gasparotto (2014), é o Google Android. É um sistema de código aberto da Google que está presente em grande parte dos smartphones hoje em dia. Outro sistema operacional com uma fatia interessante do mercado é o iOS, da Apple. Por fim, é interessante destacar o Windows Phone, que vem tendo um crescimento constante nos últimos anos (GASPAROTTO, 2014).

A maioria das aplicações móveis é criada como uma versão miniatura de aplicações para *desktop*. Contudo, segundo Ballard (2007), para que uma aplicação móvel seja bem sucedida ele tem que considera não só a portabilidade do ambiente fixo para o móvel, mas também as características que se sobrepõem parcialmente em relação às aplicações *desktop* e complementá-las, em vez de ser apenas um subconjunto dessas aplicações (DANTAS, 2009).

A grande vantagem das aplicações móveis é a capacidade que possuem de se comunicar com os dispositivos do aparelho, como câmera, acelerômetro, GPS, entre outros. Isso faz com que a aplicação fique mais completa. Por isso que é importante saber de antemão tudo o que a sua aplicação irá utilizar. O planejamento, quando construímos uma aplicação móvel, é tão ou mais importante que o código que será utilizado pela mesma. Além desses detalhes, as aplicações móveis normalmente precisam lidar com questões de aproveitamento de energia, para não gastar muita bateria, e também com a potência, para que o aparelho não superaqueça (GASPAROTTO, 2014). O Quadro 1 traz uma comparação das tecnologias dos principais sistemas operacionais disponíveis no mercado.

Quadro 1 - Tecnologia dos principais sistemas operacionais para dispositivos móveis

Sistema Operacional	Ling. de desenvol.	Tecnologia de interfaces	IDE	SO de desenvolvimento
iOS	Objective-C	Cocoa Touch	XCode	MacOS
Android	Java		Eclipse / Android Studio	Windows, Linux, MacOS
Windows Phone	C#	Metro	Visual Studio	Windows

Fonte: Gasparotto (2014).

Em ambiente móvel, de acordo com Gasparotto (2014), as regras são diferentes, não por causa da falta de normas, devido às grandes variações das necessidades dos usuários móveis. As diferentes capacidades dos celulares, dos smartphones e de outros tipos de dispositivos móveis levam a um ambiente variável. Desta forma, ainda conforme Gasparotto (2014), uma aplicação móvel consiste de:

- a) um dispositivo, um navegador, um ambiente de aplicação e suas capacidades;
- b) um usuário, usando qualquer conjunto de dispositivos, que poderia estar andando de trem, sentado em uma reunião, sentado em um restaurante, caminhando na rua ou focado em outras tarefas;
- c) uma ou mais plataformas de aplicação, que pode incluir navegadores *web*, ambiente de aplicação, mensagens, ambientes de mídia e novas capacidades que podem vir a ser disponibilizadas;
- d) uma ou mais interfaces de saída do dispositivo incluindo tela, caixa de som, IrDA, *Bluetooth*, *Wi-Fi*, dentre outros;
- e) uma ou mais interfaces de entrada do dispositivo, tais com teclado limitado, tela sensível ao toque, microfone, câmera, leitor de *Radio Frequency IDentification* (RFID), infravermelho, *Bluetooth*, *Wi-Fi*;
- f) operacionalmente um servidor que complementa a aplicação móvel e adiciona informação ou funcionalidade;
- g) interfaces entre servidor e aplicação e outras fontes de informação;
- h) uma rede com um operador sem fio que habilita algumas das tecnologias citadas, conecta o usuário a internet, vende outras aplicações e outros serviços e frequentemente define o que pode ser acoplado ou não a rede.

2.5. REPRESENTANTES COMERCIAIS

O representante comercial intermedia e facilita o relacionamento entre o produtor ou fornecedor do produto e o cliente. Como atua no processo de distribuição de produtos de diversos segmentos, a representação comercial é uma das formas mais usuais de expandir os negócios (SERVIÇO BRASILEIRO DE APOIO ÀS MICRO E PEQUENAS EMPRESAS, 2013).

Conforme Serviço Brasileiro de Apoio às Micro e Pequenas Empresas (2013), as principais funções do representante comercial são: visitar clientes e potenciais clientes, informar sobre produtos e fornecedores. Também é função de um representante realizar demonstrações de novos produtos e gerar interesse, sondar necessidades e expectativas. Desta forma oferecer soluções através dos produtos representados, fechar vendas e preencher pedidos para enviar à empresa representada, enviar relatórios sobre as negociações em andamento, pesquisar informações sobre clientes potenciais, avaliar e mensurar a satisfação do cliente enviando feedback à empresa representada e efetuar cobrança de valores quando solicitado.

Representação comercial é uma modalidade de intermediação de negócios mercantis, ou seja, os representantes comerciais têm a função de facilitar os negócios envolvendo a venda de produtos ou mercadorias de seus clientes, chamados de empresas representadas. Esta intermediação envolve de um lado as empresas representadas, indústrias e/ou empresas dedicadas ao comércio atacadistas, e de outro lado seus clientes, outras empresas atacadistas ou varejistas. Dessa forma cabe ao representante comercial fazer a ponte entre a empresa representada e seus clientes, de modo a aumentar o número de negócios entre eles (BRASIL, 2002).

Perante estas funções já conhecidas, representantes estão diretamente ligados a tecnologia. Sendo necessária para as atividades do dia-a-dia. Entretanto, tem-se a necessidade de deslocarem constantemente e por este motivo dispositivos móveis se tornam mais utilizados e são necessários para a intermediação com o cliente. Com o auxílio de aplicações móveis que, possuem tecnologias embarcadas, favoreçam o contato com o cliente tornam o atendimento rápido e eficaz.

2.6. TRABALHOS CORRELATOS

Podem-se citar como trabalhos correlatos as monografias realizadas por Thomas Alexandre Sens (SENS, 2009) e Kelly Cristina Boeck (BOECK, 2012) para conclusão do curso na universidade Regional de Blumenau que também tem a utilização das API's do Google Maps. Também tem-se como correlação o sistema Base CRM da empresa norte americana (BASE CRM, 2014).

O trabalho de Sens (2009) foi o desenvolvimento de um sistema de entrega com

otimização geográfica para uma rede de distribuidoras. O trabalho teve por objetivo a interação entre os consumidores e fornecedores, disponibilizando uma interface para manter os pedidos realizados e visualizar rotas. Este sistema calcula a melhor rota para entrega com base na distribuidora mais próxima do destino, partindo de uma origem para um ou mais destinos. Sendo possível visualizar as rotas de entrega através de mapas disponibilizados por um *Web Services*.

O sistema foi implementado utilizando as técnicas de *Model View Controller* (MVC) na linguagem *Hypertext Preprocessor* (PHP), sendo usado o *Zend Framework* e *Google Maps* API como biblioteca, algoritmos genéticos para o cálculo de melhor rota e o *MySQL* como base de dados (SENS, 2009). A Figura 1 apresenta a tela onde são visualizadas as distribuidoras mais próximas.

Figura 1 - Informações do pedido

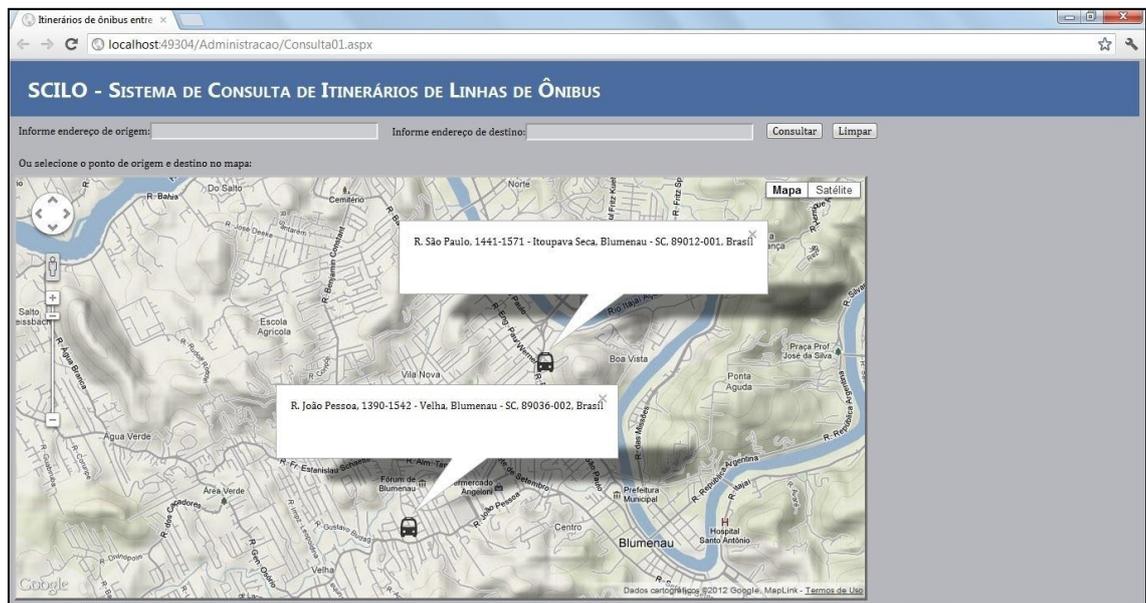
The screenshot displays a web browser window with the URL <http://www.valepizza.com/pedido-cadastro>. The page title is "Suas informações para a entrega do pedido no ValePizza.com - Mozilla Firefox". The browser's address bar shows the URL. The page content includes a progress bar with steps: 1. Endereço, 2. Restaurantes, 3. Tamanhos, 4. Sabores, 5. Adicionais, 6. Informações, 7. Finaliza. The "Informações" step is highlighted. Below the progress bar, there is a "Voltar" button. The main heading is "Informações" with the sub-heading "Informe seus dados para cadastro". Under "Seus dados", there are input fields for "Nome completo: * Thomas", "Telefone 1: * (47) 9631-9020", "Data de Nascimento: * 25/10/81", and "Telefone 2:". The "Endereço de entrega" section includes a "CEP" field with the value "89037120", a "Estado" dropdown menu set to "Santa Catarina", a "Cidade" dropdown menu set to "Blumenau", a "Bairro" dropdown menu set to "ASILO", and fields for "Complemento", "Endereço: * RUA ARTUR POLI", "Número: * 221", and "Local: * Casa" (with a note "(Casa, Apartamento, Empresa)"). A map below the address fields shows the location. On the right side, there is an "Extrato do pedido" (Order Summary) section with a yellow background. It lists: "Pizzaria Don Corleone", "Taxa de entrega ASILO: R\$ 3.50", "Pizzas: 1 Extra grande (R\$ 32.90 - 4 sabor(es)), 2 Baby (R\$ 9.90 - 2 sabor(es))", "Adicionais: 3 Coca-Cola (R\$ 5.00)", and "Total: Valor total R\$ 53.30, Pontos do pedido: 23". There are also "Anuncie aqui" buttons with dimensions "190x90px".

Fonte: Sens (2009).

Já o trabalho de Boeck (2012) foi o desenvolvimento de uma aplicação *web* para consulta de itinerários e horários de transporte público urbano, onde os itinerários são

disponibilizados para visualização no Google Maps. A aplicação permite o cadastro de linhas, horário e linhas para compor o itinerário que as linhas de ônibus atendem. O itinerário é apresentado graficamente através da API do Google Maps. O desenvolvimento da aplicação foi utilizado ASP NET, linguagem C# em conjunto com .Net Framework 4. Para as páginas de consulta utilizou-se HTML e Java Script. A Figura 2 demonstra a tela onde os itinerários são exibidos.

Figura 2 - Consulta de linhas de ônibus

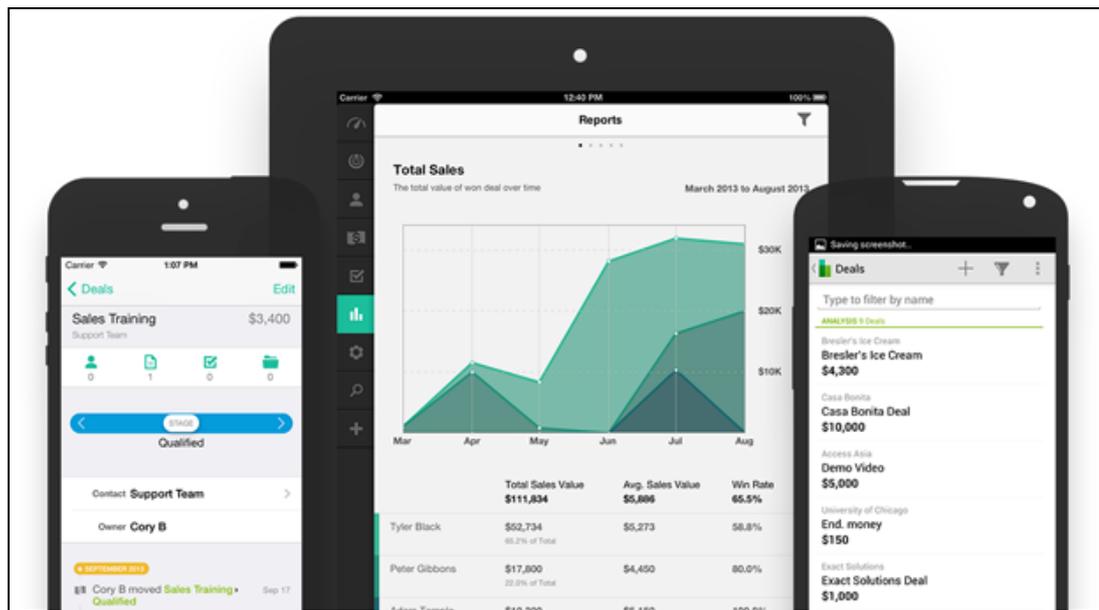


Fonte: Boeck (2012).

O sistema Base foi inicialmente desenvolvido com a fundação da empresa Base em 2009, devido as experiências frustrantes com a utilização de outras ferramentas CRM. O sistema foi desenvolvido para melhor atender as necessidades das empresas com um software mais inteligente (BASE CRM, 2014).

O Base CRM tem como foco a gestão dos atendimentos aos clientes e ao crescimento do negócio. Através do relacionamento entre o vendedor e os clientes retornando dados claros e significativos sobre os clientes através de gráficos e relatórios. A Figura 3 demonstra estas funcionalidades.

Figura 3 - Relatórios e gráficos Base CRM

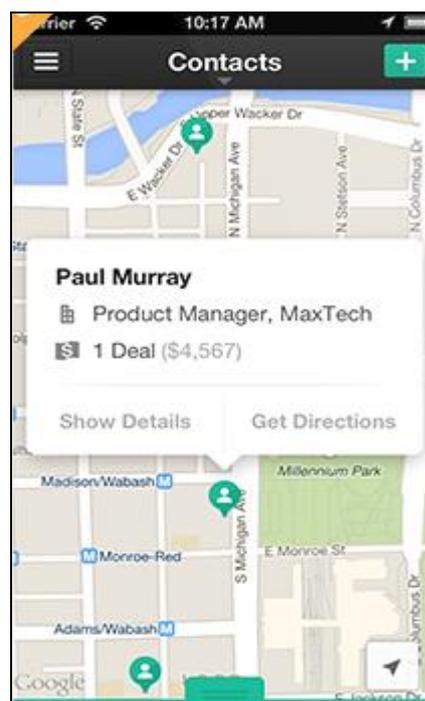


Fonte: Base CRM (2014).

Este sistema é específico para dispositivos móveis, tanto para celular quanto para *tablet* e está disponível para as plataformas Android, iOS e Windows Phone. Para teste, possui uma versão *freeware* durante quatorze dias, sendo ao final deste período necessário a compra do software disponível em três versões, Starter, Professional e Enterprise.

Na Figura 4 é demonstrado a tela do sistema onde é exibido o cliente no mapa.

Figura 4 - Cliente exibido no mapa



Fonte: Base CRM (2014).

3 DESENVOLVIMENTO DA FERRAMENTA

Nesta seção, serão abordados os levantamentos de informações, requisitos, a especificação através das sessões diagrama de casos de uso, diagrama de atividades e diagrama de classes. Em seguida serão abordados os assuntos referente a implementação do sistema de localização geográfica para representantes comerciais apresentando as técnicas e ferramentas e a operacionalidade da aplicação móvel. Por fim serão descritos os resultados e discussões.

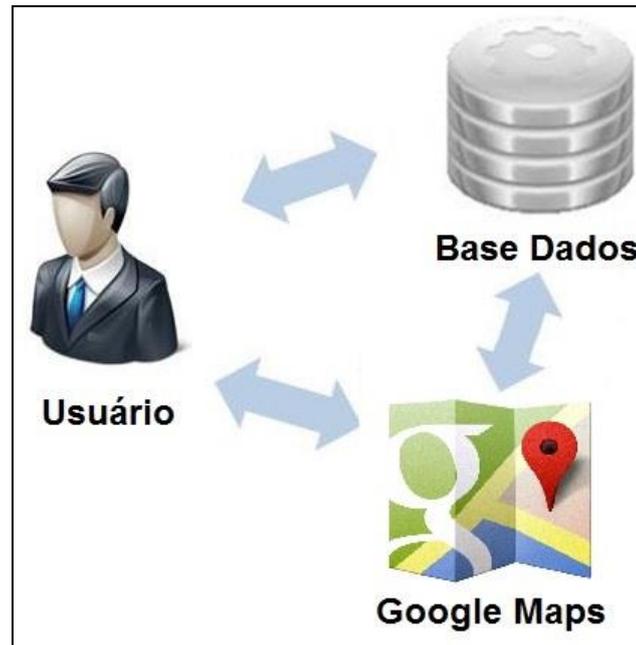
3.1 LEVANTAMENTO DE INFORMAÇÕES

Com base em informações levantadas em pesquisas na área de representação, contato informal com representantes comerciais e conforme visto nos quatro últimos parágrafos da introdução, representantes constantemente se deparam com algumas dificuldades:

- a) localizar clientes fora das cidades nas quais residem;
- b) planejar o percurso para atendimento dos clientes no dia.

Para auxiliar nas dificuldades encontradas pelos representantes, foi desenvolvida uma ferramenta com as seguintes funcionalidades: manter cadastro de clientes, criação de uma agenda para automatizar o planejamento e o cálculo da melhor rota, de acordo com a API do Google Maps, para o atendimento dos clientes agendados. O funcionamento do sistema pode ser observado de uma forma macro na Figura 5.

Figura 5 - Funcionamento do sistema



3.1.1 Requisitos principais

Com base nas dificuldades mencionadas anteriormente, no Quadro 2 são apresentados os requisitos funcionais e os requisitos não funcionais apresentados no Quadro 3, da aplicação móvel desenvolvida.

Quadro 2 - Requisitos Funcionais

REQUISITOS FUNCIONAIS		Caso de Uso
RF01	O sistema deve calcular a melhor rota entre a origem e um ou mais pontos de destino.	UC04
RF02	O sistema deve exibir a rota com base nos clientes da agenda.	UC04
RF03	O sistema deve permitir o representante manter clientes.	UC01
RF04	O sistema deve permitir o representante manter pontos de interesses no mapa.	UC03
RF05	O sistema deve calcular o tempo gasto para todas as visitas agendadas.	UC04
RF06	O sistema deve exibir pontos de interesse já cadastrados.	UC03
RF07	O sistema deve permitir o representante manter agenda.	UC02
RF08	O sistema deve exibir no mapa apenas os clientes já cadastrados.	UC04

Quadro 3 - Requisitos não funcionais

REQUISITOS NÃO FUNCIONAIS	
RNF01	Utilizar a Google Maps API v.2 para o desenvolvimento da interface para visualização de rotas.
RNF02	Implementar a ferramenta utilizando a IDE Eclipse para JAVA com as ferramentas de desenvolvimento Android.
RNF03	Utilizar o banco de dados SQLite localmente na aplicação.
RNF04	Utilizar o <i>Web Service</i> da Google para exibição das rotas.

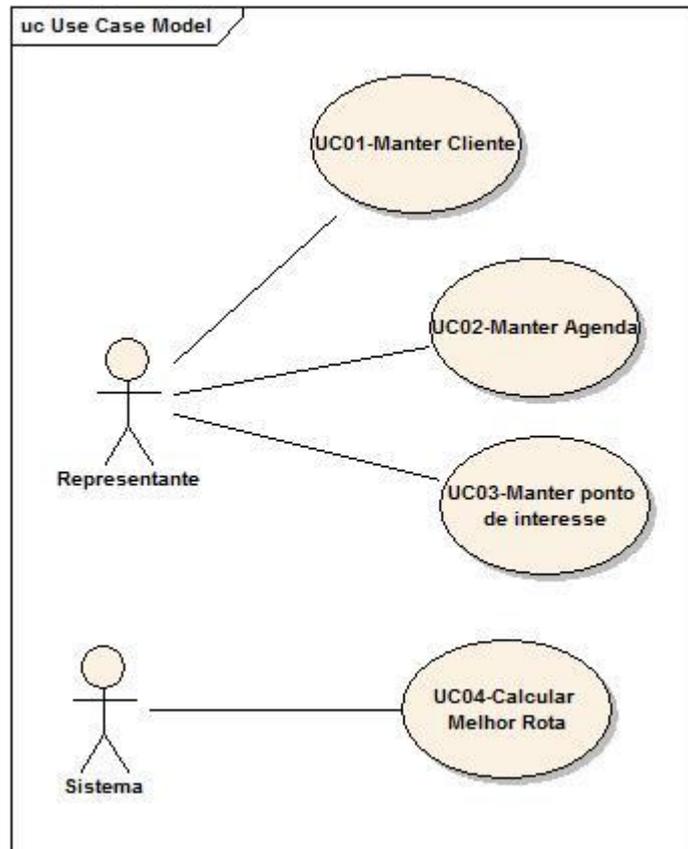
3.2 ESPECIFICAÇÃO

Para o desenvolvimento dos diagramas do projeto, foi utilizada a representação *Unified Modeling Language* (UML), com auxílio da ferramenta Enterprise Architect, sendo criado o diagrama de casos de uso, diagrama de atividade e diagrama de classes.

3.2.1 Diagrama de casos de uso

Na Figura 6 são demonstradas as ações realizadas pelos atores representante e sistema dentro da ferramenta. A descrição dos principais casos de uso é apresentada no Apêndice A.

Figura 6 - Diagrama de casos de uso



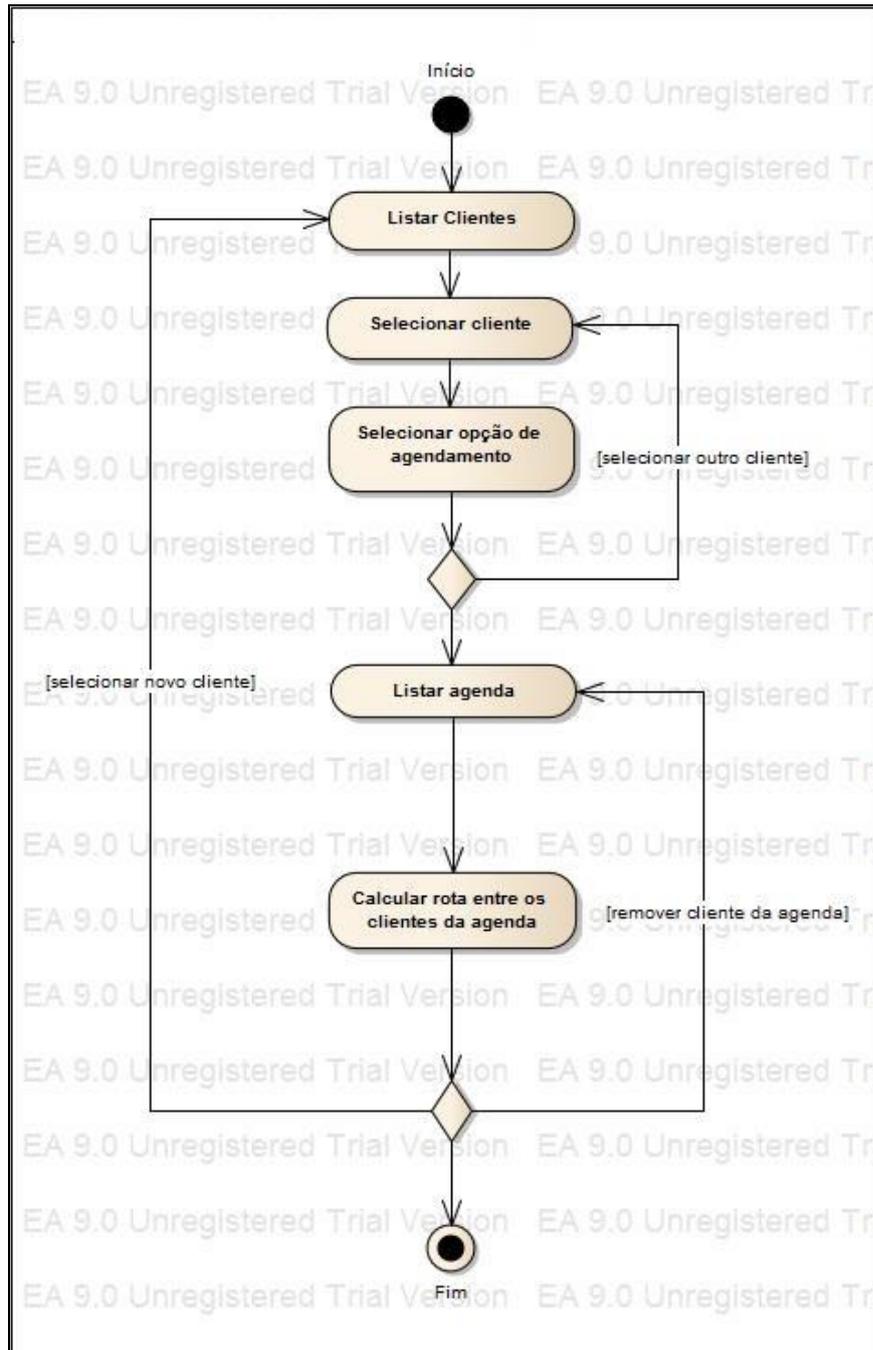
O ator representante, tem a atividade de manter o cadastro de clientes realizando cadastro, manutenção e exclusão de clientes da base de dados da ferramenta. Outra atividade, é a de manter os cadastros agendados. Neste caso de uso é possível incluir e excluir clientes na agenda. Também é atividade do ator representante cadastrar pontos de interesse no mapa para facilitar e auxiliar na localização dos clientes.

O ator sistema, após o ator representante ter realizado os cadastros de clientes na agenda, tem a possibilidade de calcular a melhor rota com base nos clientes agendados.

3.2.2 Diagrama de atividades

O diagrama de atividades apresentado na Figura 7 ilustra o processo do cálculo de melhor rota entre os clientes cadastrados na agenda do sistema. O processo da seleção de clientes pode ser executado quantas vezes forem necessárias, até que o representante opte por exibir a rota no mapa.

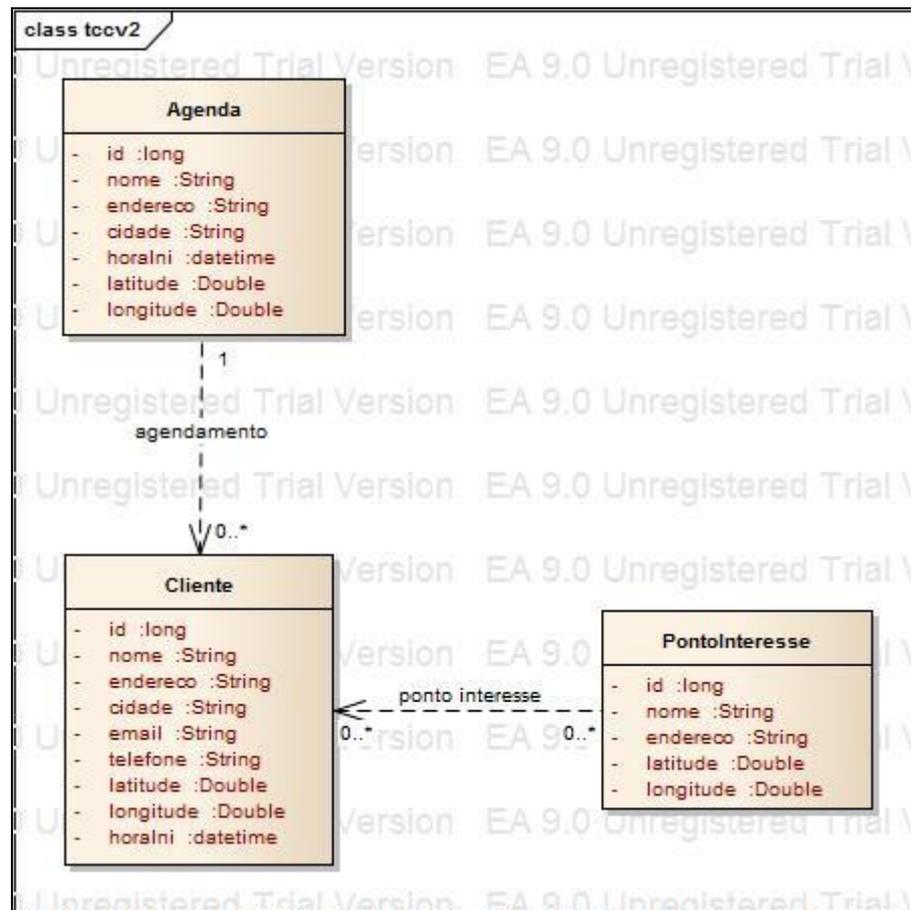
Figura 7 - Diagrama de atividades



3.2.3 Diagrama de classes

O diagrama de classes apresentado na Figura 8, exibe as principais classes encontradas no sistema com seus devidos atributos. Algumas classes foram abstraídas para uma melhor visualização do diagrama. Cada classe representa uma funcionalidade da ferramenta desenvolvida.

Figura 8 - Diagrama de classes



Na classe Agenda estão os métodos para exibir os clientes agendados pela classe Cliente e disponibilizar as opções para manter a agenda, disponibilizando funções para manutenção dos cadastros. Os clientes cadastros nesta classe serão utilizados para traçar a rota no mapa.

Na classe Cliente estão os métodos destinados a listar os cadastrados na base de dados. Tem funções que possibilitam manter os cadastros de clientes realizando, alterações nos cadastros, agendamentos e exclusão.

Na classe PontoInteresse estão os métodos para manter os pontos de interesse a serem utilizados no mapa.

3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da

implementação.

3.3.1 Técnicas e ferramentas utilizadas

Para o desenvolvimento da ferramenta foi utilizado a IDE Eclipse com as ferramentas de desenvolvimento Android e o banco de dados SQLite para a persistência de dados. Também foi utilizada a API versão 2 do Google Maps e o *Web Service* disponível pela Google para o cálculo da melhor rota. Para utilizar e testar a ferramenta, foi utilizado o emulador Android, simulando um aparelho Galaxy Nexus com 768MB de memória e processador ARMeabi-v7a. Também disponíveis na suíte de ferramentas para desenvolvimento Android.

3.3.1.1 Ferramenta de desenvolvimento Android

A ferramenta de desenvolvimento Android fornece as bibliotecas de API's e as ferramentas necessárias para o desenvolvimento, teste e depuração de aplicativos para Android. Esta ferramenta também é chamada de Android SDK. Nela estão inclusos além dos componentes essenciais, também uma versão da IDE Eclipse com o *Android Developer Tools* (ADT) embutido (ANDROID DEVELOPERS, 2013). Na Figura 9 abaixo é exibido a versão e as ferramentas inclusas no Android SDK.

Figura 9 - IDE Eclipse com *Android Developer Tools*

3.3.1.2 API Google Maps versão 2

Com a API do Google Maps versão 2 para Android, é possível adicionar mapas com base em dados do Google Maps para a aplicação. A API trabalha automaticamente com acessos aos servidores do Google Maps, realizando *download* de dados, visualização de mapas e resposta a gestos no mapa. Também é possível usar os métodos da API para adicionar marcadores, polígonos, sobreposições e ainda mudar a visão de uma área do mapa. Estes objetos auxiliam o fornecimento de informações adicionais para os locais e permite a interação do usuário com o mapa (IMOBILIS, 2013).

A API versão 2 do Google Maps para Android é distribuída como parte do Google Play services SDK e possui um novo sistema de gerenciamento de chaves. Onde chaves existentes de um aplicativo Android Google Maps v.1 não irão funcionar com a API v.2. A chave do Google Maps API é baseada em um pequeno formulário de certificado digital do aplicativo, conhecido como *SHA-1 fingerprint*. A *fingerprint* é uma sequência de texto único, gerado a partir do algoritmo comumente usado, a *SHA-1 hashing*, sendo única e exclusiva para o dispositivo onde será desenvolvida a aplicação.

O código da Figura 10 abaixo exhibe o comando para, na versão 2, inicializar o mapa na aplicação.

Figura 10 - Inicializa mapa

```
private void inicilizaMapa() {
    if (googleMap == null) {
        googleMap = ((MapFragment) getFragmentManager().findFragmentById(
            R.id.map)).getMap();

        LatLng latLng = new LatLng(latitude, longitude);

        Map<String, LatLng> agendaMapa = carregaAgendaMapa();
        new RotaAsyncTask(this, googleMap).execute(agendaMapa);

        // checa se o mapa foi criado com sucesso ou não
        if (googleMap == null) {
            Toast.makeText(getApplicationContext(),
                "Que pena! Não é possível criar o mapa",
                Toast.LENGTH_SHORT).show();
        }
    }
}
```

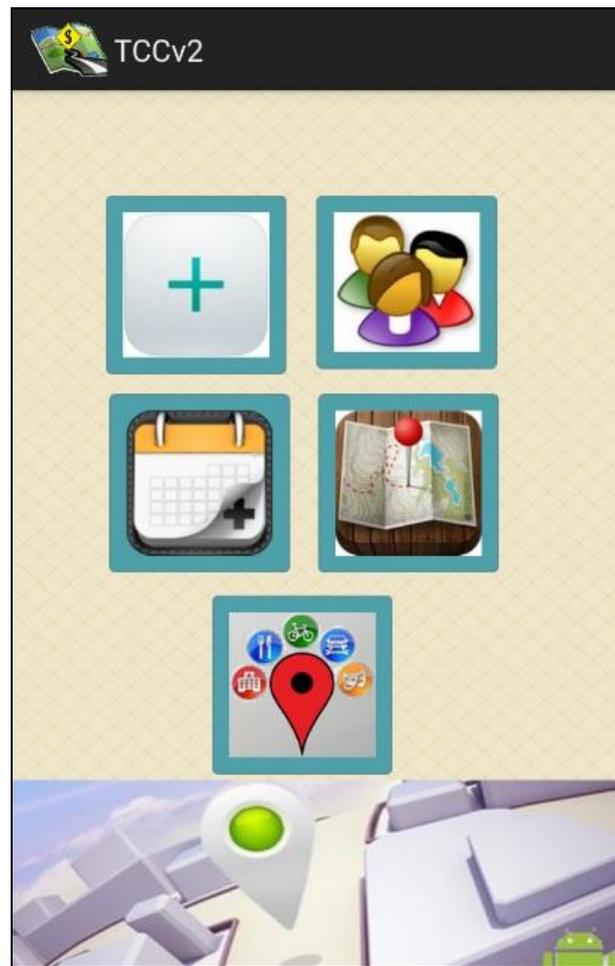
3.3.2 Operacionalidade da implementação

Nesta subseção serão descritas as telas do sistema e suas funcionalidades. Para cada tela apresentada, será demonstrado trechos de código fonte de rotinas desenvolvidas.

Para exemplificar, será descrito o processo desde o cadastro dos clientes até a exibição da melhor rota. Todos os dados dos cadastros realizados são fictícios, não referenciando pessoas da vida real.

Inicialmente, ao abrir a aplicação, é aberta diretamente a tela do menu principal da ferramenta sem a necessidade de *login*. Nesta tela o usuário pode escolher qual função deseja iniciar para iniciar o planejamento dos atendimentos: cadastrar cliente, cadastrar um ponto de interesse, consultar a relação de clientes, verificar a agenda e ver a rota para os clientes agendados. Os ícones do meu principal são ilustrados com a função que realizam. A Figura 11 apresenta a tela do menu principal.

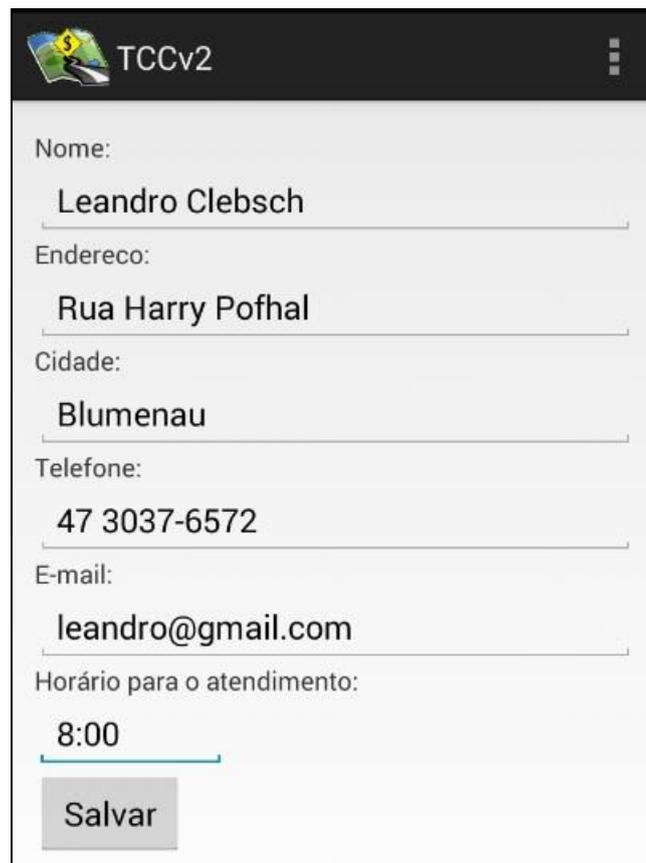
Figura 11 - Menu principal



Caso o cliente ainda não seja cadastrado na base de dados, o representante irá selecionar a opção adicionar cliente, que é representado pelo botão em formato de “mais”. No botão ilustrado por três bonecos é possível visualizar a listagem dos clientes cadastrados da base de dados, sendo possível realizar as alterações em seus cadastros. Para consultar os clientes agendados, é necessário clicar no botão ilustrado por uma agenda. O botão ilustrado por um mapa executa as funções para exibir o mapa com a rota entre os clientes agendados. Por fim, é possível cadastrar pontos de interessa clicando sobre o botão ilustrado por um marcador do Google Maps.

Na Figura 12, visualiza-se a tela de cadastro de cliente. Através de um formulário, são solicitados os dados de, nome, endereço, cidade, telefone, *e-mail* e o horário em que o cliente pode ser atendido. Ao informar estes dados, o representante deve clicar no botão salvar para gravar o cadastro realizado.

Figura 12 - Formulário de cadastro de cliente



Nome:
Leandro Clebsch

Endereço:
Rua Harry Pofhal

Cidade:
Blumenau

Telefone:
47 3037-6572

E-mail:
leandro@gmail.com

Horário para o atendimento:
8:00

Salvar

No momento em que o representante pressiona o botão “Salvar”, o sistema também converte o endereço em pontos geográficos, latitude e longitude, vinculando ao cadastro. O código da Figura 13 abaixo, mostra como é realizado o processo de conversão.

Figura 13 - Código para converter endereço em coordenadas

```

local = (txtCidade.getText().toString() + "," + txtEndereco.getText().toString());

// converter endereço em coordenadas geográficas
Geocoder geocoder = new Geocoder(this);
List<Address> enderecos;
try {
    enderecos = geocoder.getFromLocationName(local, 1);

    Address add = enderecos.get(0);
    double latitude = add.getLatitude();
    double longitude = add.getLongitude();

    String loc = String.valueOf(latitude + "," + longitude);
    Toast.makeText(this, loc, Toast.LENGTH_LONG).show();

    String loc = String.valueOf(latitude + "," + longitude);
    Toast.makeText(this, loc, Toast.LENGTH_LONG).show();
} catch (IOException e) {
    e.printStackTrace();
    Toast.makeText(getApplicationContext(), "Endereço não encontrado!", Toast.LENGTH_SHORT).show();
}

```

No código demonstrado, é passado para o *List* chamado *enderecos* o endereço da *String* denominada *local*, realizando a conversão pela função *geocoder* realizado na linha sete do código. Nas linhas dez e onze, são adicionadas a latitude e longitude coletadas, na tabela de clientes.

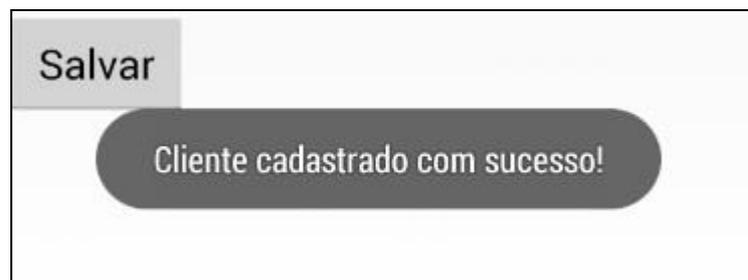
Após a conversão, é exibida uma mensagem com as coordenadas geográficas coletadas pela conversão. A Figura 14 demonstra a mensagem que é exibida.

Figura 14 - Coordenadas convertidas



Em seguida, é exibida a mensagem de confirmação do cadastro na base de dados, conforme mostrado na Figura 15.

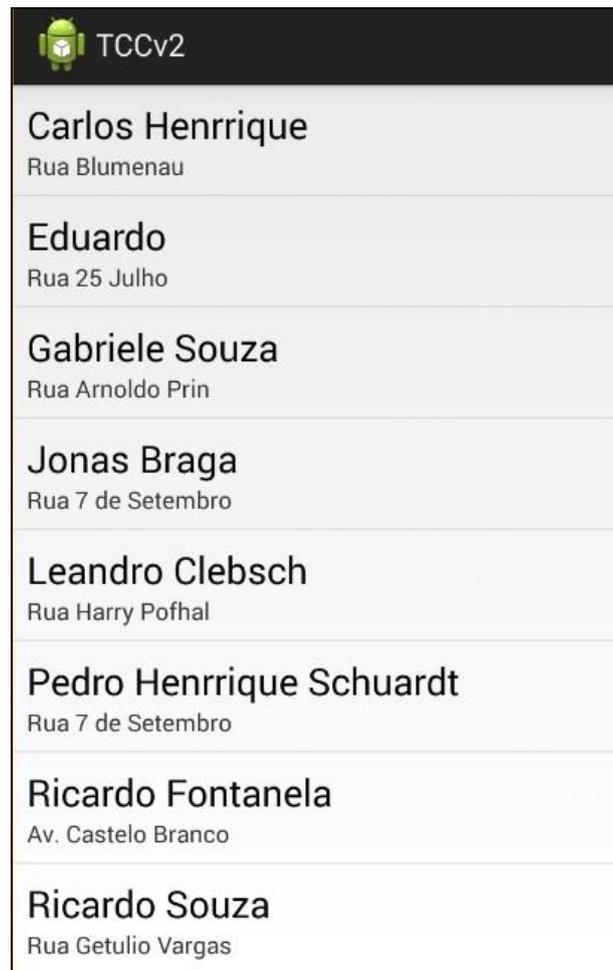
Figura 15 - Cadastro realizado



Após o cadastro ser realizado, o sistema retorna para o menu inicial, podendo ser selecionada a função desejada novamente.

Para consultar a lista de clientes o representante deve clicar no botão representado por três bonecos. A Figura 16 apresenta a tela de consulta de clientes. Nesta tela será possível efetuar o processo de edição, exclusão e agendamento de um cliente clicando sobre ele.

Figura 16 - Lista de clientes



A Figura 17 mostra o código executado para listar os clientes no visor.

Figura 17 - Método que gera a lista de clientes no visor

```
private List<Map<String, String>> listarClientes() {
    ArrayList<Map<String, String>> list = new ArrayList<Map<String, String>>();

    SQLiteDatabase db = banco.getReadableDatabase();

    Cursor cursor = db.rawQuery("SELECT _id, nome, latitude FROM clientes ORDER BY nome", null);
    cursor.moveToFirst();

    for (int i = 0; i < cursor.getCount(); i++) {
        Map<String, String> map = new HashMap<String, String>();
        map.put("_id", cursor.getString(0));
        map.put("nome", cursor.getString(1));
        map.put("latitude", cursor.getString(2));
        list.add(map);

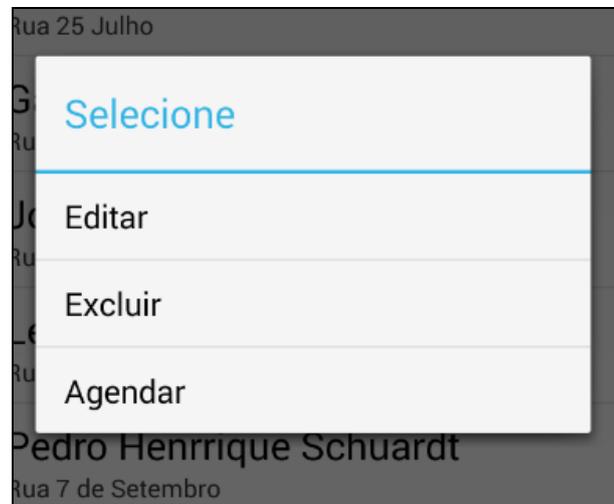
        cursor.moveToNext();
    }
}
```

O código demonstrado, na linha sete é realiza a consulta dos clientes na base de dados passando para a *String* denominada *map* iniciada na linha três.

Ao clicar sobre um cliente, é exibido o menu conforme demonstrado na Figura 18,

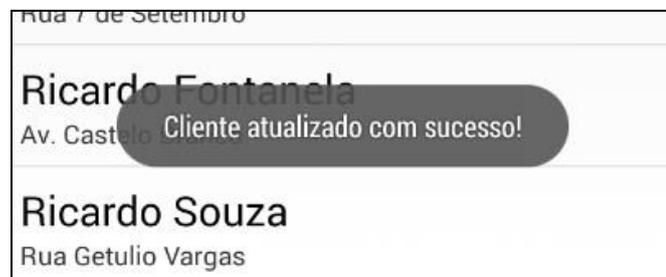
possibilitando realizar as ações de edição, exclusão e agendamento.

Figura 18 - Menu interação da tela de clientes



Quando selecionada a opção editar, o sistema redireciona para a tela de cadastro anteriormente apresentada. Nela é permitido alterar qualquer dado do cliente. Após as alterações o representante deve clicar em salvar, sendo exibida a mensagem conforme a Figura 19.

Figura 19 - Mensagem de confirmação da atualização



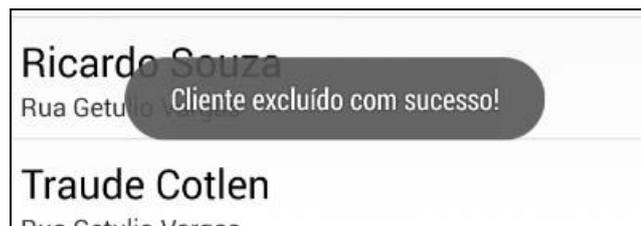
Na tela de consulta de clientes, também é possível selecionar a opção para excluir um registro da tabela de clientes. Quando selecionada, o sistema solicita a confirmação da exclusão, conforme ilustrado na Figura 20, caso contrário será retornado à listagem de clientes.

Figura 20 - Confirmação de exclusão



O registro selecionado será excluído, sendo exibida a mensagem de confirmação demonstrada na Figura 21.

Figura 21 - Mensagem de confirmação da exclusão



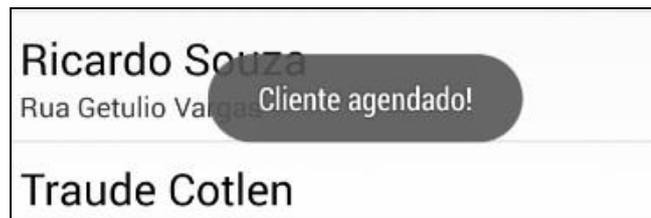
Pela tela de consulta de clientes, também é feito o agendamento dos clientes. Para isto deve-se clicar sobre a opção “Agendar”. Será solicitado a confirmação do agendamento conforme a mensagem da Figura 22.

Figura 22 - Confirmação de agendamento



O agendamento será realizado e uma mensagem, conforme a Figura 23, será exibida confirmando o agendamento.

Figura 23 - Mensagem de cliente agendado



Para realizar o agendamento, é feito uma cópia do registro selecionado na tela de clientes, para a tabela da agenda. Parte do código é ilustrado na Figura 24 que, demonstra a rotina para o processo.

Figura 24 - Método que copia cliente para a tabela da agenda

```

SQLiteDatabase db = banco.getWritableDatabase();
String where[] = { selecionado.getString("_id") };
Cursor cursor = db.rawQuery(
    "SELECT _id, nome, endereco, cidade, latitude, longitude FROM clientes where _id= ?", where);
cursor.moveToFirst();

for (int i = 0; i < cursor.getCount(); i++) {
    ContentValues content = new ContentValues();
    content.put("_id", cursor.getString(0));
    content.put("nome", cursor.getString(1));
    content.put("endereco", cursor.getString(2));
    content.put("cidade", cursor.getString(3));
    content.put("latitude", cursor.getString(4));
    content.put("longitude", cursor.getString(5));
    db.insert("agenda", null, content);
}

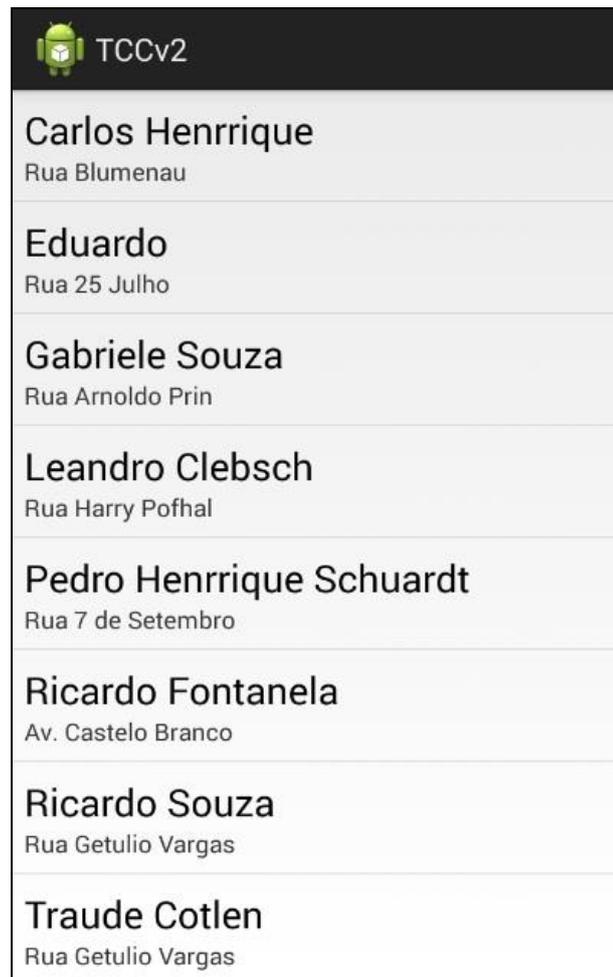
```

No código demonstrado acima, para adicionar o cliente na agenda, é realizado a consulta do cliente selecionado na tabela de clientes, passando cada valor dos campos da tabela *clientes* para a tabela *agenda*.

Toda a funcionalidade da tela de consulta de clientes foi demonstrada anteriormente e para prosseguir é necessário retornar ao menu principal.

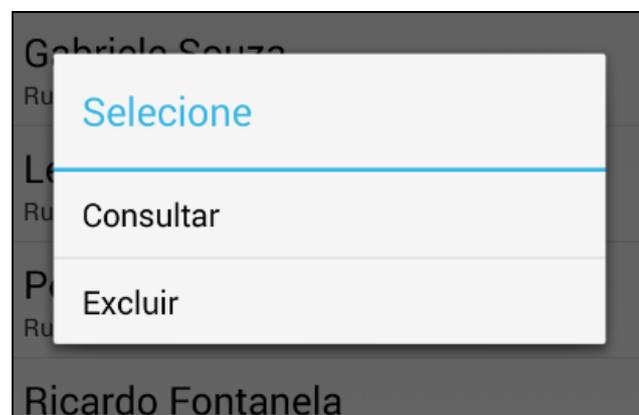
Após os clientes serem cadastrados e devidamente agendados, no menu principal é possível consultar os clientes agendados. Para esta função, no menu principal, deve-se clicar sobre o botão ilustrado por um calendário. A Figura 25 apresenta a tela de consulta da agenda.

Figura 25 - Tela da agenda



Na tela da agenda, também é possível realizar alterações nos cadastros. Ao pressionar sobre um cliente agendado, é exibido o menu conforme demonstrado na Figura 26, exibindo a interação disponível.

Figura 26 - Menu do agendamento selecionado



Ao selecionar a opção “Consultar”, o sistema redirecionará para a tela de cadastro de cliente anteriormente apresentada. Nela será possível além de visualizar os dados, também alterar se necessário. Ao final podem ser salvas as alterações, ou simplesmente retornar a tela

anterior.

Também poderá ser selecionada a opção “Excluir” onde, será exibida a mensagem para confirmar a ação. Caso contrário, será retornado para a listagem dos agendamentos. A mensagem é ilustrada na Figura 27.

Figura 27 - Confirmação de exclusão



Ao ser confirmada a exclusão, será exibida a mensagem de confirmação da exclusão demonstrada na Figura 28.

Figura 28 - Mensagem de confirmação da exclusão



Retornando ao menu inicial, após terem sido cadastrados e agendados todos os clientes desejados, será possível traçar a rota com o melhor caminho de acordo com a API do Google Maps para atendimento dos clientes com base no horário e preferência cadastrado para o cliente. Com base neste, é organizada a ordem dos atendimentos.

Também será possível cadastrar pontos de interesse no mapa. Para isto, deve-se clicar no ícone com o marcador do Google Maps. A Figura 29 demonstra a tela de cadastro de ponto de interesse onde, deve ser informado o nome do local, o endereço e a cidade.

Figura 29 - Cadastro de ponto de interesse

The screenshot shows a mobile application interface with a dark header containing a logo and the text 'TCCv2'. The main content area is a light gray form with three text input fields. The first field is labeled 'Nome:' and contains the text 'Posto Ipiranga'. The second field is labeled 'Endereco:' and contains 'Rua Antonio da Veiga, 122'. The third field is labeled 'Cidade:' and contains 'Blumenau'. Below the fields is a gray button with the text 'Salvar'.

Quando um ponto de interesse é cadastrado na base de dados, também é realizado o processo de conversão do endereço em ponto geográfico. A Figura 30 apresenta o código que realiza as operações.

Figura 30 - Método que cadastra ponto de interesse

```
@Override
public void onClick(View v) {

    SQLiteDatabase db = banco.getWritableDatabase();

    ContentValues content = new ContentValues();
    content.put("nome", txtNome.getText().toString());
    content.put("endereco", txtEndereco.getText().toString());
    content.put("cidade", txtCidade.getText().toString());

    local = (txtEndereco.getText().toString() + "," + txtCidade.getText().toString());

    // converter endereço em coordenadas geográficas

    // Geocoder geocoder = new Geocoder(this, Locale.getDefault());
    Geocoder geocoder = new Geocoder(this);
    List<Address> enderecos;
    try {
        enderecos = geocoder.getFromLocationName(local, 1);

        Address add = enderecos.get(0);
        double latitude = add.getLatitude();
        double longitude = add.getLongitude();

        content.put("latitude", latitude);
        content.put("longitude", longitude);

        String loc = String.valueOf(latitude + "," + longitude);
        Toast.makeText(this, loc, Toast.LENGTH_LONG).show();
    } catch (IOException e) {
        e.printStackTrace();
    }

    if (this.id < 0) {
        db.insert("pontoInteresse", null, content);
        Toast.makeText(PontoInteresseActivity.this, "Ponto de interesse cadastrado!", Toast.LENGTH_SHORT).show();
    }
}
```

No método da Figura 30, para converter o endereço em ponto geográfico, são passados

os valores da *String* com nome *local*, para o *List enderecos*. Por fim é coletado o ponto geográfico pela função *geocoder* da API do Google Maps.

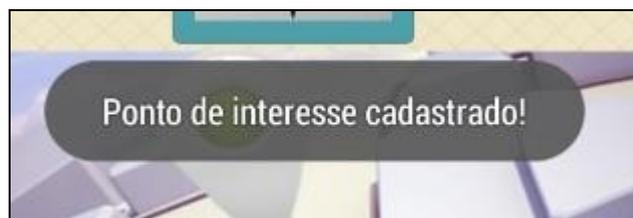
Quando o cadastro é realizado, uma mensagem com os pontos geográficos convertidos é exibida. A Figura 31 demonstra a mensagem.

Figura 31 - Coordenada do ponto de interesse



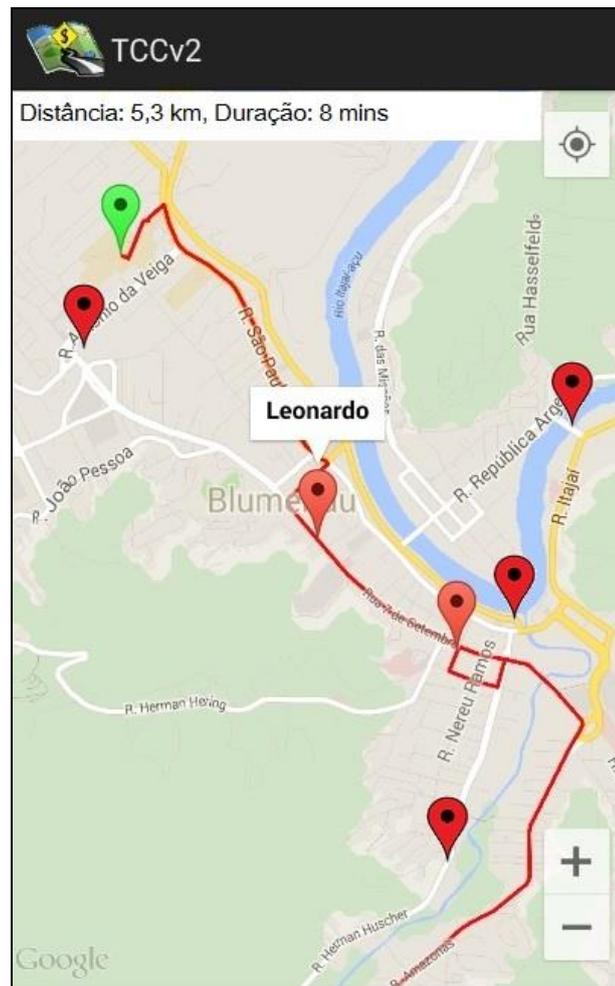
Em seguida a tela é redirecionada para o menu principal sendo exibida a mensagem de confirmação do cadastro do ponto de interesse na base de dados, conforme ilustrado na Figura 32. Ao final, podem ser cadastrados novos pontos de interesse.

Figura 32 - Confirmação cadastro ponto de interesse



Para visualizar a rota, deve-se clicar no botão ilustrado por um mapa no menu principal. Após ser selecionado, dará início ao processo de carregamento do mapa, busca do local atual do representante e cálculo da melhor rota de acordo com a API do Google Maps, entre os clientes agendados. Toda rota terá como ponto inicial a localização atual do representante que será representada com um marcador na cor verde. Em seguida será exibido uma barra na parte superior da tela com a informação da distância e o tempo para percorrer a rota selecionada. Também serão adicionados os pontos de interesse cadastrados. A Figura 33 demonstra a tela do mapa com as funcionalidades.

Figura 33 - Melhor rota



Quando pressionado sobre um ponto exibido na tela, é possível visualizar o cliente a qual este ponto referência e qual o ponto de interesse.

Ao iniciar o processo da tela do mapa, primeiramente é realizado a inicialização do mapa em tela, em seguida é traçada a rota entre os pontos e exibido a distância e tempo para o trajeto calculado pela API do Google Maps. O código da Figura 34 mostra o método para carregar o mapa na tela.

Figura 34 - Método inicializa do mapa

```

private void inicilizaMapa() {
    if (googleMap == null) {
        googleMap = ((MapFragment) getFragmentManager().findFragmentById(
            R.id.map)).getMap();

        LatLng latLng = new LatLng(latitude, longitude);

        Map<String, LatLng> agendaMapa = carregaAgendaMapa();
        new RotaAsyncTask(this, googleMap).execute(agendaMapa);

        // checa se o mapa foi criado com sucesso ou não
        if (googleMap == null) {
            Toast.makeText(getApplicationContext(),
                "Que pena! Não é possível criar o mapa",
                Toast.LENGTH_SHORT).show();
        }
    }
}

```

No método da Figura 34, o mapa é inicializado pela função *MapFragment* da API do Google Maps, chamando o método *carregaAgendaMapa*.

Após o mapa ser exibido em tela, são carregados os clientes da agenda no mapa em forma de pontos. O código que realiza este processo é demonstrado na Figura 35.

Figura 35 - Método que carrega clientes agendados no mapa

```

private Map<String, LatLng> carregaAgendaMapa() {

    Map<String, LatLng> mapa = new LinkedHashMap<>();

    SQLiteDatabase db = banco.getReadableDatabase();

    Cursor cursor = db.rawQuery("SELECT nome, latitude, longitude FROM agenda",
        null);
    cursor.moveToFirst();

    for (int i = 0; i < cursor.getCount(); i++) {
        mapa.put(cursor.getString(0), new LatLng(cursor.getDouble(1), cursor.getDouble(2)));
        cursor.moveToNext();
    }

    return mapa;
}

```

No método demonstrado acima, é passado para a *String* a lista de clientes realizada pela consulta na tabela *agenda* colocando no mapa os clientes da consulta pelo laço de repetição até chegar ao final dos resultados consultados.

Por fim é iniciada a classe para gerar a rota entre os clientes agendados, direcionando a visualização do mapa para a localização atual do representante. O método que executa este

processo é mostrado na Figura 36.

Figura 36 - Método para traçar a rota

```

@Override
protected void onPostExecute(Void result) {
    super.onPostExecute(result);
    PolylineOptions options = new PolylineOptions().width(5).color(Color.RED).visible(true);

    boolean inicial = true;
    for (Map.Entry<String, Route> entry : mapaRotas.entrySet()) {
        Route rota = entry.getValue();
        String chave = entry.getKey();

        if (inicial) {
            LatLng coordenada = rota.getPoints().get(0);
            mapView.addMarker(new MarkerOptions().position(coordenada).title("Inicial"));
            CameraPosition posicaoCamera = new CameraPosition.Builder().target(coordenada).zoom(15).build();

            mapView.animateCamera(CameraUpdateFactory.newCameraPosition(posicaoCamera));
            inicial = false;
        }

        LatLng ultimo = null;
        for (LatLng latlng : rota.getPoints()) {
            ultimo = latlng;
            options.add(latlng);
        }
        mapView.addMarker(new MarkerOptions().position(ultimo).title(chave));
    }

    mapView.addPolyline(options);
}

```

O método da Figura 36, inicialmente define o tipo de linha que será traçada pela função *PolylineOptions* e traça a rota pela função *Route* da API, passando os valores da variável *mapView*.

Após a rota ser exibida e clientes serem atendidos é permitido retornar ao menu principal para realizar o processador que for desejado.

3.4 RESULTADOS E DISCUSSÃO

A aplicação móvel desenvolvida neste trabalho atingiu as expectativas propostas, possibilitando ao representante comercial visualizar a melhor rota retornada pela API do Google Maps entre os clientes agendados, apresentando uma estimativa do tempo que será gasto para o deslocamento entre os clientes. A aplicação móvel atende a funcionalidades específicas que, em relação aos trabalhos correlatos e ao sistema similar, não são atendidas especificamente com a indicação da melhor entre os clientes desejados.

Este projeto foi desenvolvido na plataforma Android, para dispositivos móveis, direcionada a representantes comerciais integrada com a API do Google Maps e GPS. Diferente dos trabalhos correlatos que não são destinados a dispositivos móveis, entretanto utilizam a API do Google Maps para a representação dos mapas. Com o crescimento do uso de dispositivos móveis aumentando constantemente, aplicações que não destinam-se a estes tipos de dispositivos, não se tornam tão usuais. Uma vez que o acesso deve ser feito por um local fixo específico. O sistema Base CRM por sua vez, disponibiliza interfaces para dispositivos móveis e computadores. O Quadro 4 relaciona as principais características entre os trabalhos correlatos, sistema similar e a aplicação desenvolvida.

Quadro 4 - Comparativo entre aplicações

Sistema	Sistema Móvel	Cads.	Web Service	Agenda	Integs.	Exibe Rotas	API Google Maps	Banco de dados local	Consultas	Otimiza processos
Pedidos de Pizza		X	X		X	X	X		X	X
Itinerários trans. Público		X					X	X	X	X
Base CRM	X	X		X	X		X	X	X	X
Sistema desenvolvido	X	X		X		X	X	X	X	X*

*Neste caso, é realizado a otimização pela API do Google Maps. Muitas vezes não será a melhor solução.

Por se tratarem de aplicações específicas, algumas características não são correlatas a aplicação desenvolvida. Com isto, é identificado que, a solução de “Pedidos de pizza” tem por finalidade na localização da distribuidora mais próxima, não sendo possível realizar planejamentos, nem a visualização em um dispositivo móvel. A solução para “Itinerários de transporte público” não é destinada para dispositivos móveis e tem por objetivo identificar o local que um ônibus está em um determinado momento, não exibindo a rota, nem uma agenda para consulta. No sistema Base CRM, é exibido um mapa com a rota para o cliente porém, a rota é apenas para o cliente selecionado não sendo possível visualizar a rota para mais de um cliente.

Entretanto, as ferramentas também possuem características em comum como a possibilidade de cadastros dentro da aplicação, utilização da API do Google Maps para exibir o mapa e a otimização de um processo que anteriormente não existiam, ou, foram automatizados.

Durante o processo de desenvolvimento da solução, foi identificado que, as soluções disponíveis no mercado não disponibilizam a rota entre os clientes a serem atendidos, assim como uma solução de baixo custo para representantes comerciais. Com isto, notou-se que a aplicação móvel desenvolvida possui uma vantagem em relação aos sistemas apresentados anteriormente.

4 CONCLUSÕES

A busca por ferramentas que otimizam as tarefas do dia-a-dia é algo presente em todos os ramos de atividades empresariais. Com isto a otimização de processos e recursos, através do desenvolvimento de sistemas, proporciona ganhos significativos de produtividade e inovação para as atividades cotidianas. Oferecer ferramentas que otimizam processo são imprescindíveis para empresas que desejam reduzir custos operacionais, passando a ser um diferencial em um mercado atual cada vez mais competitivo.

Os objetivos propostos neste trabalho foram alcançados, com a criação de uma solução Android com orientação geográfica para representantes comerciais com a utilização do Google Maps e GPS, desta forma, auxilia no planejamento das visitas para otimizar recursos.

Com o cálculo da melhor rota retornado pela API do Google Maps sendo traçada no mapa, entre os clientes adicionados na agenda, possibilita percorrer o caminho mais curto, de acordo com o método do Google Maps, até o cliente que será atendido. Desta forma são reduzidos os gastos com combustível e o tempo do deslocamento, aumentando o tempo dedicado para o atendimento ao cliente. Ainda para auxiliar na localização, pontos de interesse são adicionados ao mapa referenciando locais estratégicos, reduzindo ainda mais o tempo gasto com a localização do cliente.

Através da utilização das ferramentas de desenvolvimento Android dentro da IDE Eclipse, juntamente com o banco de dados SQLite, foi possível integrar a API do Google Maps na aplicação. Foi utilizada a versão 2 da API por possuir recursos atualizados para o desenvolvimento e melhor compatibilidade com as versões atuais do sistema operacional Android do dispositivo móvel.

Durante o processo de desenvolvimento da aplicação, foram encontrados problemas com o ambiente de desenvolvimento, onde, as bibliotecas da API do Google Maps não eram encontradas pelo Eclipse. A referência da pasta onde as bibliotecas estão instaladas não era encontrada, sendo necessário removê-las e adicionar novamente. Em outros casos foi necessário recriar o projeto. Também tiveram que ser corrigidos problemas de compatibilidade da API versão 2 do Google Maps com a versão 2.3.3 do sistema operacional Android. Para a solução deste problema, foi retirada a compatibilidade da aplicação desenvolvida com esta versão do sistema operacional Android.

Todo o processo de pesquisa, desenvolvimento e a conclusão deste trabalho proporcionaram a aplicação de todo o conhecimento adquirido durante o curso. Desde a

concepção da ideia, análise do problema, desenvolvimento, até os componentes visuais que formam a interface para o usuário. Além do conhecimento adquirido no decorrer do curso, houve a oportunidade de aprimorar o conhecimento em relação a API do Google Maps que já era conhecida, mas foi necessário buscar materiais para aplicação das funcionalidades implementadas na aplicação desenvolvida e de novas ferramentas para o desenvolvimento. Com isto puderam ser atendidas as necessidades dos representantes para, principalmente, otimizar o deslocamento entre os clientes.

4.1 EXTENSÕES

O sistema apresentado neste trabalho auxilia representantes no planejamento para o deslocamento entre os clientes agendados, traçando o melhor caminho a ser percorrido entre eles. Diante deste cenário podem ser sugeridas as seguintes extensões:

- a) integração com sistemas de força de vendas e sistemas de gestão da empresa representada;
- b) histórico de atendimentos realizados por período;
- c) demonstração de gráficos, em estilo *dashboard*, para medir os clientes atendidos em um determinado período, quilômetros percorridos, tempo gasto com viagens, valor gasto com deslocamento;
- d) a medida que o nome da rua for digitado, sugerir nomes de ruas que, possivelmente podem ser a desejada;
- e) navegação por voz;
- f) desenvolver um método próprio para encontrar o melhor caminho entre os pontos no mapa.

REFERÊNCIAS

AIR FORCE, Los Angeles Base. **FACTSHEETS: GPS Advanced Control Segment (OCX)**. Los Angeles, 2011. Disponível em: <<http://www.losangeles.af.mil/library/factsheets/factsheet.asp?id=18676>>. Acesso em: 6 abr. 2013.

ALEXANDROW, Catherine. **The Story of GPS**. Arligton, 2013. Disponível em: <<http://www.darpa.mil/workarea/downloadasset.aspx?id=2565>>. Acesso em: 13 abr. 2013.

ALMEIDA, Regina A. de; GUERRERO, Ana L.; FIORI, Sérgio R. **Geografia e cartografia para o turismo**. São Paulo: IPSIS, 2007.

ANDROID DEVELOPERS. **Get the Android SDK**. Califórnia, 2013. Disponível em: <<https://developer.android.com/sdk/index.html>>. Acesso em: 05 jul. 2014.

AUGUSTIN, Iara, et al. **Requisitos para o projeto de aplicações móveis distribuídas**. VII Congreso Argentino de Ciencias de la Computación, 2001.

BALLARD, B. **Designing the Mobile User Experience**. Little Springs Design, Inc. USA: Wiley, 2007.

BASE CRM. **The intelligent sales productivity platform**. [S.l.], 2014. Disponível em: <<https://getbase.com/tour/>>. Acesso em: 05 out. 2014.

BOECK, Kelly Cristina. **Aplicação WEB para consulta de itinerários de transporte público com visualização no Google Maps**. 2012. 81 f, il. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2012.

BRASIL, Código civil. Lei nº 4886/65, Alterada pela lei 8420/92: **Contrato de Representação Comercial**. Brasília, 2002.

CARVALHO, Alexandre. **O geoprocessamento como recurso para análise das ocorrências em linhas de ônibus urbano em Belo Horizonte – um estudo de caso**. 2002. 55 f. Monografia (Especialização) – Departamento de Cartografia, Universidade Federal de 78 Minas Gerais, Belo Horizonte.

CRIARENET. **O que fazemos**. São Paulo, 2012. Disponível em: <<http://www.criarenet.com/isp/o-que-fazemos/mobile-application/qual-plataforma/>>. Acesso em: 07 nov. 2014.

DANA, Peter H. **Global Positioning System Overview**. Texas, 1999. Disponível em: <<http://www.colorado.edu/geography/gcraft/notes/gps/gps.html#PosVelTime>>. Acesso em: 14 abr. 2013.

DANTAS, Valéria Lelli Letão. **Requisitos para Testes de aplicações móveis**. 2009. 30 f. Mestrado e Doutorado em Ciência da Computação - Universidade Federal do Ceará - Departamento de Computação, Fortaleza.

DAVID, W. Allan. **The Science of Timekeeping. Hewlett Packard. Archived from the original**. Englewood, 1999. Disponível em: <http://www.allanstime.com/Publications/DWA/Science_Timekeeping/TheScienceOfTimekeeping.pdf>. Acesso em: 14 abr. 2013.

GASPAROTTO, Henrique Machado. **Aplicações Móveis: Nativas ou Web?** Santa Maria, 2014. Disponível em: <<http://www.devmedia.com.br/aplicacoes-moveis-nativas-ou-web/30392>>. Acesso em: 9 nov. 2014.

GOOGLE. **A nossa História em Detalhes**. Mountain View, 2013a. Disponível em: <<http://www.google.com.br/about/company/history/>>. Acesso em: 14 abr. 2013.

GOOGLE. **Mapping your way** – Google Official Blog. Mountain View, 2013b. Disponível em: <<http://googleblog.blogspot.com.br/2005/02/mapping-your-way.html>>. Acesso em: 14 abr. 2013.

GUIMARÃES, Gleyser. **A história do sistema operacional Android**. Campina Grande. 2013. Disponível em: <http://www.dsc.ufcg.edu.br/~pet/jornal/agosto2013/materias/historia_da_computacao.html>. Acesso em: 10 mar. 2014.

HISTORIA DE TUDO. **História do GPS**. [S.l.], 2013. Disponível em: <<http://www.historiadetudo.com/gps.html>>. Acesso em: 01 abr. 2013.

INSTITUTO NACIONAL DO LIVRO (BRASIL). **Manual do representante do INL**. Brasília, D.F : INL, 1972. 103p.

IMOBILIS. **Tutorial: Introdução** – API Google Maps V2 Android. Ouro Preto, 2013. Universidade Federal de Ouro Preto – Departamento de Computação, Minas Gerais. Disponível em: <<http://www.decom.ufop.br/imobilis/?p=2546>>. Acesso em: 12 out. 2014.

LECHETA, Ricardo R. **Google Android: Aprenda a criar aplicações para dispositivos móveis com o Android SDK2**. Ed. São Paulo Novatec, 2010. 608 p,il.

MONTEIRO, João Bosco. **Google Android. Crie aplicações para celulares e tablets**. São Paulo. Casa do código, 2013.

SERVIÇO BRASILEIRO DE APOIO ÀS MICRO E PEQUENAS EMPRESAS. **Expand a empresa com o apoio de representantes comerciais.** Brasília, 2013. Disponível em: <<http://www.sebrae.com.br/customizado/aceso-a-mercados/distribua-seus-produtos/representante-comercial>>. Acesso em: 16 abr. 2013.

SENS, Thomas Alexandre. **Sistema de pedidos de pizza para otimização de rotas no Google Maps.** 2009. 55 f, il. Trabalho de Conclusão de Curso - (Graduação em Ciência da Computação) - Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau, 2009.

APÊNDICE A – DESCRIÇÃO DOS CASOS DE USO

Este Apêndice apresenta a descrição dos principais casos de uso descritos na seção de especificação deste trabalho. No Quadro 5 é apresentado o caso de uso Manter clientes.

Quadro 5 - Caso de uso Manter clientes

UC01 – Manter clientes

Ator: Representante.

Objetivo: O representante comercial pode cadastrar, alterar, excluir ou agendar um cadastro de cliente da base de dados.

Pré-condição: Cliente cadastrado.

Pós-condição: Cadastro do cliente alterado.

Pós-condição: Cliente agendado.

Pós-condição: Cadastro do cliente excluído.

Cenário Principal:

1. Representante acessa a tela através do menu principal a opção “Clientes”;
2. O sistema a lista de clientes cadastrados;
3. Representante seleciona o cliente desejado;
4. Sistema exibe um menu com as opções para Editar, Excluir e Agendar;
5. Representante seleciona a opção “Agendar”;
6. Sistema solicita confirmação;
7. O representante confirma;
8. O sistema realiza o agendamento do cliente.

Cenário Alternativo 01:

No passo 4 o representante seleciona a opção “Editar”:

1. O sistema direciona para a tela de cadastro de clientes;
2. Representante realiza as alterações e salva;
3. O sistema atualiza o cadastro com as informações alterada e retorna para o passo 3 do cenário principal.

Cenário Alternativo 02:

No passo 4 o representante seleciona a opção “Excluir”:

1. O sistema solicita confirmação;
2. Representante confirma;
3. O sistema exclui o registro do cliente da base de dados e retorna para o passo 3 do cenário principal.

O Quadro 6 apresenta o caso de uso Manter agenda.

Quadro 6 - Caso de uso Manter agenda

UC02 – Manter agenda

Ator: Representante.

Objetivo: O representante pode consultar ou excluir os clientes selecionados para visita.

Pré-condição: Cliente cadastrado.

Pré-condição: Cliente agendado.

Pós-condição: Cliente disponível para traçar a rota.

Pós-condição: Cliente excluído do agendamento.

Cenário Principal:

1. O representante seleciona no menu principal a opção “Agenda”;
2. O sistema exibe listagem de clientes agendados;
3. O representante seleciona o cliente desejado;
4. O sistema exibe menu com as opções “Consulta” e “Excluir”;
5. O representante seleciona a opção excluir;
6. O sistema ao representante a confirmação;
7. O representante confirma;
8. O sistema exclui o registro de cliente da agenda e retorna para a tela com a listagem de clientes agendados.

Cenário Alternativo 01:

No passo 4 o representante seleciona a opção “Consultar”:

1. O sistema redireciona para a tela de cadastro de clientes;
2. O representante clica em retornar;
3. O sistema redireciona para a tela da agenda e retorna para o passo 2 do cenário principal.

O Quadro 7 apresenta o caso de uso Manter ponto de interesse.

Quadro 7 - Caso de uso Manter ponto de interesse

UC03 – Manter ponto de interesse

Ator: Representante.

Objetivo: É possível marcar pontos de interesse no mapa para auxiliar na referência e localização.

Pré-condição: Cliente cadastrado.

Pós-condição: Ponto de interesse cadastrado.

Cenário Principal:

1. O sistema exibe formulário para cadastro na tela de cadastro de clientes;
2. O representante informa o endereço do ponto;
3. O sistema grava o cadastro na base de dados.

O Quadro 8 apresenta o caso de uso Cálculo de melhor rota.

Quadro 8 - Caso de uso Cálculo de melhor rota

UC04 – Cálculo de melhor rota

Ator: Sistema.

Objetivo: Através dos critérios de cálculo de melhor rota da API do Google Maps e dos clientes agendados, o sistema retorna com a melhor rota entre os pontos.

Pré-condição: Clientes cadastrados.

Pré-condição: Clientes agendados.

Pós-condição: Exibe rota entre os clientes agendados.

Cenário Principal:

1. O representante seleciona a opção “Mapa” no menu principal;
2. O sistema carrega os clientes da agenda no mapa, traçando a rota entre eles;

Cenário Alternativo 01:

No passo 2 o representante opta por agendar um novo cliente:

1. O representante retorna para o menu inicial e seleciona a opção “Clientes”;
2. O sistema exibe a relação de clientes;
3. O representante seleciona o cliente desejado;

4. O sistema exibe menu com as opções Edita, Excluir e Agendar;
5. O representante seleciona a opção “Agendar”;
6. O sistema solicita confirmação;
7. O representante confirma;
8. O sistema realiza o agendamento;
9. O representante retorna ao menu principal e retorna ao passo 1 do cenário principal.

Cenário alternativo 02:

No passo 2 o representante opte por excluir um agendamento:

1. O representante retorna ao menu principal e seleciona a opção “Agenda”;
2. O sistema exibe os clientes agendados;
3. O representante seleciona o cliente desejado;
4. O sistema exibe menu com as opções Consulta e Excluir;
5. O representante seleciona a opção “Excluir”;
6. O sistema solicita confirmação;
7. O representante confirma;
8. O sistema exclui o agendamento e retorna ao passo 2 do cenário alternativo 02;
9. O representante volta para o menu principal e retorna ao passo 1 do cenário principal.