

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CURSO DE ENGENHARIA DE COMPUTAÇÃO**

HANDREY EMANUEL GALON

**SISTEMA DE RASTREAMENTO E CONTROLE DE RECURSOS DE
UM VEÍCULO UTILIZANDO UM SMARTPHONE ANDROID**

TRABALHO DE CONCLUSÃO DE CURSO

**PATO BRANCO
2014**

HANDREY EMANUEL GALON

**SISTEMA DE RASTREAMENTO E CONTROLE DE RECURSOS DE UM
VEÍCULO UTILIZANDO UM SMARTPHONE ANDROID**

Trabalho de Conclusão do Curso de Engenharia de Computação, apresentado à UTFPR como requisito parcial para obtenção do título de Engenheiro de Computação.

Orientador: Prof. Me. Robison Cris Brito

**PATO BRANCO
2014**

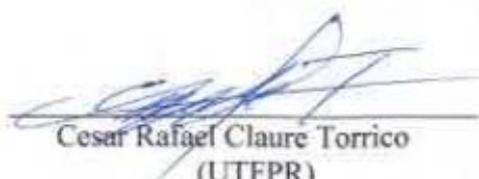


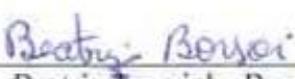
TERMO DE APROVAÇÃO

Às 16h30 do dia 14 de agosto de 2014, na sala V107, UTFPR, Câmpus Pato Branco, reuniu-se a banca examinadora composta pelos professores Robison Cris Brito (Orientador), Fábio Favarim e Cesar Rafael Claire Torrico para avaliar o trabalho de conclusão de curso com o título **Sistema de rastreamento e controle de recursos de um veículo utilizando um smartphone**, do aluno **Handrey Emanuel Galon**, matrícula 1065084, do curso de Engenharia de Computação. Após a apresentação o candidato foi arguido pela banca examinadora. Em seguida foi realizada a deliberação pela banca examinadora que considerou o trabalho aprovado.


Robison Cris Brito
Orientador (UTFPR)


Fábio Favarim
(UTFPR)


Cesar Rafael Claire Torrico
(UTFPR)


Beatriz Verezinha Borsoi
Coordenador de TCC


Marco Antonio de Castro Barbosa
Coordenador do Curso de
Engenharia de Computação

RESUMO

GALON, Handrey E. Sistema de rastreamento e controle de recursos de um veículo utilizando um smartphone Android. 2014. 79 f. Monografia de Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Pato Branco, 2014.

Desde os primórdios a humanidade buscou maneiras de se localizar no globo terrestre de forma rápida e eficaz. A sociedade moderna também exige que as pessoas estejam conectadas constantemente afim de que as informações sejam compartilhadas de forma rápida. No presente trabalho é proposto um protótipo envolvendo hardware e software, visando o desenvolvimento de uma plataforma de monitoramento e rastreamento de veículo. O software para uso do usuário foi desenvolvida usando a plataforma Android, visando seu uso nos *smartphones*. Para o acionamento e recuperação dos dados presentes no veículo, foi utilizado um microcontrolador TIVA TM4C123G, conectado a um módulo GSM SIM900, e a um módulo GPS NEO 6M para recuperação da posição do veículo. A tecnologia utilizada para transmissão de dados entre o software e o hardware foi SMS. Para os testes, foi utilizado um motor DC com caixa de redução para simulação do acionamento dos vidros elétricos, um servo motor Tower Pro SG-5010 para simulação do acionamento das travas elétricas, e um sensor PIR ligado a uma speaker para simulação de um alarme. Assim, o software desenvolvido para *smartphone* Android pôde controlar a distância os motores que simulavam os vidros e travas elétricas, assim como recebia uma notificação caso o sensor PIR fosse acionado. Já o módulo GSM conectado ao microcontrolador também enviava a cada intervalo de tempo pré-programado as coordenadas de longitude/latitude capturado pelo módulo GPS.

Palavras-chave: Android. Rastreamento. Veículo. Microcontrolador.

ABSTRACT

GALON, Handrey E. Tracking and control of resources of a vehicle using an Android Smartphone. 2014. 79 f. Monografia de Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Pato Branco. Pato Branco, 2014.

Since the dawn, the humanity sought ways to locate on the globe quickly and effectively. Modern society also requires that people are constantly connected in order that information is shared quickly. This paper proposes a prototype involving hardware and software, for the development of a platform for vehicle tracking. The user's software was developed using the Android platform, aiming its use in smartphones. To drive and data recovery in the vehicle was used a microcontroller TIVA TM4C123G, connected to a SIM900 GSM module and a NEO 6M GPS module to recover the vehicle position. The technology used to transmit data between the software and the hardware was SMS. For the tests, it was used a DC motor with reduction gearbox to drive the simulation of electric windows, a servo motor Tower Pro SG-5010 to drive the simulation of electric locks, and PIR sensor connected to a speaker to simulate an alarm. Thus, the developed software for Android smartphone could control, from distance, the engines that simulated the windows and power locks as well as receiving a notification if the PIR sensor was triggered. Already the GSM module connected to the microcontroller also sent to each preset time interval the coordinates of longitude/latitude captured by the GPS module.

Keywords: Android. Tracking. Vehicle. Microcontroller.

LISTA DE FIGURAS

Figura 1 – O globo terrestre e suas linhas de latitude e longitude.....	17
Figura 2 – Analogia para determinação de coordenadas usando sinal GPS em 2D ...	19
Figura 3 – HTC T-Mobile, primeiro aparelho com Android.....	24
Figura 4 – Diagrama de blocos de um microcontrolador	27
Figura 5 – Sascar APP	28
Figura 6 – Netraster APP.....	29
Figura 7 – <i>My BMW Remote</i>	30
Figura 8 – Bloqueador BL Free 906	30
Figura 9 – Esquema básico de funcionamento do sistema	32
Figura 10 – Tiva LaunchPad	36
Figura 11 – SIM900 S2-1040S	37
Figura 12 – Módulo GPS NEO-6M.....	38
Figura 13 – Sensor PIR HC-SR501	39
Figura 14 – Speaker	40
Figura 15 – Servo Motor Tower Pro SG-5010	41
Figura 16 – Motor DC	42
Figura 17 – L293B.....	42
Figura 19 – Android Virtual Device 4.2.2.....	46
Figura 20 – Diagrama de caso de uso do sistema.....	52
Figura 21 – Diagrama de Classes	57
Figura 22 – Diagrama Entidade-Relacionamento	57
Figura 24 – Esquema de montagem dos componentes do protótipo	62
Figura 25 – Tela Inicial do aplicativo Android	64
Figura 26 – Lista de posições	66
Figura 27 – Mapa referente uma posição do veículo.....	71
Figura 28 – Protótipo do projeto.....	72

LISTA DE QUADROS

Quadro 1 – Vantagens e desvantagens da tecnologia GPS	21
Quadro 2 – Formato dos dados do protocolo GPRMC	22
Quadro 3 – Requisito ativar alarme	48
Quadro 4 – Requisito ativar trava elétrica	48
Quadro 5 – Requisito levantar vidro elétrico.....	49
Quadro 6 – Requisito desativar alarme	49
Quadro 7 – Requisito desativar trava	49
Quadro 8 – Requisito baixar vidro elétrico	49
Quadro 9 – Requisito rastreamento	50
Quadro 10 – Requisito executar ações no veículo.....	50
Quadro 11 – Requisito verificar estado do alarme.	50
Quadro 12 – Casos de uso	51
Quadro 13 – Casos de uso – Ativar alarme	53
Quadro 14 – Casos de uso – Desativar alarme	53
Quadro 15 – Casos de uso – Levantar vidros elétricos.....	54
Quadro 16 – Casos de uso – Descer vidros elétricos.....	54
Quadro 17 – Casos de uso – Ativar travas elétricas	55
Quadro 18 – Casos de uso – desativar travas elétricas	55
Quadro 19 – Casos de uso – Verificar posição do veículo.....	55
Quadro 20 – Casos de uso – Executar as ações no veículo	56
Quadro 21 – Casos de uso – Verificar estado do alarme.....	56
Quadro 22 – Ação a ser executada de acordo com a SMS recebida pelo módulo GSM63	

LISTAGENS DE CÓDIGO

Listagem 1 - Criação do banco de dados SQLite	65
Listagem 2 - Recuperando conteúdo do SMS	65
Listagem 3 - Inserção das informações no banco de dados.....	66
Listagem 4 - <i>Activity</i> responsável por apresentar a lista de posições para o usuário .	67
Listagem 5 - Permissões para trabalhar com SMS em Android.....	68
Listagem 6 - Classe que trata o recebimento de SMS	68
Listagem 7 - Comandos para o envio de SMS	69
Listagem 8 - Adicionando a chave no projeto para utilizar a Google Maps API.....	69
Listagem 9 - Permissões de usuário anexadas ao manifesto	70
Listagem 10 - Arquivo de definição do layout para apresentar o mapa na aplicação	70
Listagem 11 - <i>Activity</i> Mapa.....	70

SUMÁRIO

1 INTRODUÇÃO.....	10
1.1 Problemas e motivações	11
1.2 Justificativa.....	13
1.3 Objetivo geral	14
1.4 Objetivos específicos	14
1.5 Estrutura do trabalho	15
2 FUNDAMENTAÇÃO TEÓRICA	16
2.1 Geoinformática	16
2.1.1 Coordenadas Geográficas	17
2.1.2 GPS.....	18
2.1.3 Protocolo NMEA 0183.....	21
2.2 A rede de comunicação das operadoras de telefonia móvel.....	23
2.3 Plataforma de desenvolvimento para dispositivos móveis Android.....	24
2.4 Recursos eletrônicos presentes nos veículos	25
2.5 Microcontroladores.....	26
2.6 Trabalhos Relacionados.....	27
2.6.1 Sascar APP	27
2.6.2 Netraster APP	28
2.6.3 My BMW Remote	29
2.6.4 BL Free 906.....	30
2.6.5 Sistema de alarme integrando GPS e celular.....	31
3 METODOLOGIA.....	32
3.1 Rastreamento do veículo	33
3.2 Acionamento dos recursos do veículo	34
4 RECURSOS DE HARDWARE E SOFTWARE	35
4.1 Recursos de hardware.....	35
4.1.1 Microcontrolador	35
4.1.2 Módulo GSM/GPRS SIM900 S2-1040S.....	36
4.1.3 Módulo GPS NEO-6M	38
4.1.4 Alarme veicular	39

4.1.5 Servo motor	40
4.1.6 Motor DC.....	41
4.2 Recursos de software	42
4.2.1 IDE Energia	43
4.2.2 IDE Eclipse.....	44
4.2.3 SQLite.....	45
4.2.4 Android SDK.....	45
4.2.5 Google Maps API	46
5 PROJETO DESENVOLVIDO	48
5.1 Projeto de software	48
5.1.1 Levantamento de requisitos	48
5.1.2 Diagramas de casos de uso	50
5.1.3 Diagrama de Classes.....	56
5.1.4 Diagrama Entidade-Relacionamento	57
5.2 Projeto de hardware	58
5.2.1 Aplicativo	58
5.2.2 Arquitetura do sistema.....	58
5.2.3 Funcionalidade do sistema	60
6 RESULTADOS E TESTES DE VALIDAÇÃO	61
6.1 Desenvolvimento do protótipo	61
6.2 Programação do Microcontrolador	63
6.3 Aplicativo Android	64
6.3.1 Classe Principal	64
6.3.2 Classe responsável pela lista de posições	66
6.3.3 Trabalhando com SMS em Android	67
6.3.4 Trabalhando com mapas em Android.....	69
6.4 Testes e Verificação do Projeto	71
7 CONCLUSÃO.....	73
7.1 Contribuições.....	73
7.2 Trabalhos Futuros	73
REFERÊNCIAS	75

1 INTRODUÇÃO

A era da comunicação iniciou com a necessidade das pessoas manterem contato umas com as outras sem se preocuparem com a distância. Hoje, devido ao constante avanço tecnológico, o ato de comunicar-se já não se trata mais de apenas conversar com alguém que está a milhares de quilômetros. A questão agora é totalmente voltada para a praticidade e a utilidade dessas tecnologias de comunicação.

Antigamente, os aparelhos celulares serviam basicamente para realizar e receber chamadas telefônicas. Hoje, eles possuem muitas outras funções e podem ser utilizados até mesmo como um pequeno computador. Com a evolução dos aparelhos, surgiram os *smartphones* que já agregam vários recursos que até então só existiam em computadores, como armazenamento de dados, processamento, entrada e saída de informações através de uma interface com o usuário, entre outros. Assim como os aparelhos celulares, seus sistemas operacionais também evoluíram (ANDROID, 2013a). Um dos principais existentes hoje é o Android, este mantido pela Google e já conquistou 79% do mercado de *smartphones* (KLEINA, 2014).

O diferencial do Android em relação aos outros sistemas operacionais destinado a dispositivos móveis é o fato de o mesmo ser elaborado em código aberto, possibilitando a criação de soluções para estes dispositivos baseados na plataforma Linux. A Google também disponibiliza o Android SDK (*Software Development Kit*), que é a ferramenta que permite que os desenvolvedores elaborem as aplicações a partir de um dispositivo virtual para dispositivos móveis (ANDROID, 2013b).

Os *smartphones* podem se comunicar com outros dispositivos remotos usando a rede mundial de computadores, seja utilizando o acesso via rede da operadora, ou via pontos de acesso como Wi-Fi, ou ainda, é possível a comunicação a curta distância com outros dispositivos sem fio, utilizando, por exemplo, o protocolo Bluetooth. Estes dispositivos remotos podem ser computadores, impressoras, servidores ou até mesmo outros dispositivos eletrônicos, como por exemplo, um microcontrolador.

Um microcontrolador é um sistema computacional completo, no qual estão incluídos uma CPU (*Central Processor Unit*), memória de dados e programa, um sistema de *clock*, portas de I/O (*Input/Output*), além de outros possíveis periféricos (DENARDIN, 2013). Entretanto, apesar de possuir um conjunto reduzido de instruções, este é capaz de controlar inúmeros recursos, sendo uma ótima alternativa ao uso de computadores pessoais no

desenvolvimento de tarefas específicas, tais como acionar algum recurso ou ler dados de algum sensor.

O uso destes microcontroladores está cada vez mais comum hoje em dia, podendo este ser comunicado com outros dispositivos através de uma conexão física, por meio de cabos, ou por conexões sem fio, como Wi-Fi ou Bluetooth (com o auxílio de *shields* que aumentam as funcionalidades de um microcontrolador).

Os microcontroladores disponíveis hoje no mercado oferecem uma ampla gama de opções de periféricos, incluindo ADC (Conversor Analógico Digital), PWM (Modulação por Largura de Pulso), além de interfaces I2C (*Inter-Integrated Circuit*), SPI (*Serial Peripheral Interface*), que são meio de conectar periféricos ao microcontrolador.

Com tantos recursos, é possível o desenvolvimento de uma aplicação prática e útil fazendo o uso da comunicação entre um *smartphone* e o acionamento de dispositivos específicos usando microcontrolador, controlando os recursos de um veículo à distância. Este processo pode ser feito em duas vias: controlando alguns recursos do veículo (acionamentos) ou lendo as informações sobre o veículo (por exemplo, sua posição a partir de um aparelho GPS (*Global Position System*)).

Com o aumento da frota de veículos no Brasil, cerca de 119% nos últimos dez anos, representado um total de aproximadamente 64,8 milhões de veículos no país (DENATRAN, 2010), a utilização do controle a distância dos recursos de um veículo pode ser um diferencial para as montadoras e vendedoras, ainda mais se for possível tal controle a baixo custo.

A venda dos *smartphones* também cresceu cerca de 78% em 2012, um total de 16 milhões de unidades somente no Brasil (TOZETTO, 2012). A ideia de integrar estes aparelhos com os veículos é bastante útil, para fins de comodidade (acionamento a distância) ou até mesmo segurança.

Este trabalho irá especificar as funcionalidades de uma plataforma de rastreamento veicular, que alia sistemas de informação de implantação ágil para aplicação de rastreamento através de um *smartphone* a um hardware, que dispõe de um receptor de coordenadas GPS e envia dados remotamente pela rede GSM, bem como o controle do alarme, travas elétricas e vidros elétricos do veículo.

1.1 Problemas e motivações

A sociedade moderna vive hoje a era digital ou era da informação, em que os computadores e máquinas são inseridos em nossa conjuntura para automatizar e auxiliar todas

as tarefas que antes eram manuais. Não só no ponto de vista dos processos, mas essa associação homem-máquina chegou ao âmbito pessoal e trouxe consigo uma mudança no modo como a sociedade se relaciona e se comunica (LEMOS et al., 2009).

Nestes novos tempos vivemos a era da conectividade na qual toda informação deve estar na palma da nossa mão e disponível 24 horas por dia. Foi devido a essa evolução que as novas soluções de hardware e software obtiveram a mobilidade como ponto forte, visando simplificar para o usuário a utilização dos serviços remotos e/ou *online*, sem que ele se preocupe em onde está o dispositivo, já que essa necessidade é suprida pelas mais diferentes interfaces de comunicação, como mensagens de texto e o acesso a Internet.

Serviços como rastreamento via GPS nos dispositivos móveis se tornaram alvos bastante explorados, em especial, pelo fato desses dispositivos possuírem mecanismos de comunicação com a rede (Internet) e com outros dispositivos, seja via mensagem de texto ou mensagens multimídia. Essa necessidade de se ter a comunicação com mecanismos capazes de realizar remotamente uma transmissão de dados abriu espaço para as soluções de telemetria. Este aspecto ganhou grande enfoque devido a difusão das redes sem fio que facilitaram o estabelecimento da comunicação, impulsionando a fabricação de hardwares mais robustos, com preços menores e eficientes quanto ao consumo de energia.

O desenvolvimento também se tornou significativo no setor de geoprocessamento, impulsionando a expansão dos Sistemas Globais de Navegação por Satélite, dos Sistemas de Informação Geográfica (GIS) e do mapeamento *web*. Com a ajuda dos provedores de mapas como o Google Maps, foi possível automatizar a produção de documentos cartográficos e, conseqüentemente, os projetos na área de georreferenciamento (ALVEZ, 2011).

Devido a todos estes avanços, no campo de desenvolvimento dos sistemas de informação, pode-se notar a constante preocupação com custos, tempo de produção e manutenção dos softwares produzidos.

Como já existem sistemas semelhantes dentro de grandes montadoras de veículos, procura-se com o presente projeto proporcionar um sistema a um custo acessível e que possa ser implantado em qualquer veículo, realizando o controle e acionamentos de recursos do mesmo através de um *smartphone* e registrar suas últimas posições.

O sistema proposto é um diferencial nas questões referentes à segurança, pois o usuário será avisado sobre as atividades do seu veículo e poderá tomar providências em relação às mesmas. Algumas das vantagens do sistema serão:

- Rastreabilidade, em que o usuário sempre poderá verificar o atual posicionamento do seu veículo;

- Mercado, pois o sistema poderá ser implementado em qualquer veículo com um custo reduzido;
- Comodidade, pois o usuário não precisará estar necessariamente no veículo para realizar alguma ação;
- Satisfação pessoal (status), hoje em dia as pessoas estão sempre buscando estar inseridos no mundo moderno, acompanhando as novas tecnologias e as tendências do mercado.

1.2 Justificativa

Os *smartphones* tornaram-se a melhor alternativa para que uma pessoa esteja conectada com outras e com a rede mundial de computadores. Estes aparelhos móveis possuem cada vez mais funcionalidades, aumentando com isso a comodidade das pessoas. Este trabalho visa o desenvolvimento de uma solução em Android capaz de controlar algumas funcionalidades de um veículo.

Atualmente, não há muitas maneiras para verificar se o alarme do seu veículo está acionado. A maneira mais comum é através de alguma indicação luminosa no veículo ou através do som emitido pela sirene do alarme quando este está acionado. Geralmente, o controle das travas elétricas de portas e dos vidros elétricos de um veículo é feito manualmente pelo usuário (pressionando as teclas correspondentes dentro do veículo), e em alguns casos automáticas (via acionamento de alarme ou quando o veículo atinge uma determinada velocidade).

Outra funcionalidade útil seria rastrear a posição do veículo através de um GPS, o que seria muito interessante, caso acontecesse deste ser roubado. Desta maneira, percebe-se que um aplicativo que contemple o controle dessas funcionalidades, ainda com a vantagem de ter baixo custo, seria de grande utilidade aos usuários além de um diferencial para as montadoras e empresas vendedoras de automóveis.

Desta forma, pretende-se desenvolver um sistema que realize o controle dessas funcionalidades através de um *smartphone*. Este sistema tem como função permitir o controle do alarme, trava elétrica e vidro elétrico das portas e caso o alarme do automóvel seja disparado, o sistema deverá enviar um aviso para o celular através de uma mensagem de texto (SMS – *Short Message Service*), o usuário por sua vez poderá desativá-lo e/ou reativá-lo novamente se desejado, bem como verificar a posição atual do veículo através do GPS.

Optou-se pelo uso de mensagens de texto pelo pequeno número de caracteres enviados e recebido pelo serviço proposto neste trabalho, que fica dentro do limite dos 160 caracteres suportados pelo SMS, além de hoje, mesmo com a popularização da Internet entre os dispositivos móveis, ainda existem mais dispositivos com suporte a SMS (habilitados para enviar e receber mensagens) do que habilitados para acessar a rede mundial de computadores – Internet.

Usando apenas SMS, também se elimina a necessidade de um servidor HTTP para comunicação usando a Internet dos dispositivos móveis, através da rede GPRS. Este servidor seria responsável por receber a requisição ou solicitação do dispositivo móvel e enviar para o microcontrolador presente no carro.

1.3 Objetivo geral

Construir um sistema que permite o controle dos recursos de um automóvel a distância, assim como recuperação de dados deste, utilizando *smartphone*, um microcontrolador e a rede da operadora.

1.4 Objetivos específicos

1. Analisar os sistemas semelhante;
2. Implementar um software que permita a conexão do microcontrolador com o módulo GSM e o módulo GPS;
3. Implementar um software para o *smartphone* realizar a comunicação com o microcontrolador;
4. Realizar a integração do hardware e do software para compor a plataforma;
5. Permitir que o usuário verifique se o alarme do veículo foi disparado, bem como se o mesmo está ativado/desativado.
6. Permitir que o usuário verifique as posições em que seu veículo esteve, as quais são enviadas a cada intervalo de tempo pré-determinado para o dispositivo móvel.
7. Apresentar a utilidade do sistema e futuras implementações;

1.5 Estrutura do trabalho

Capítulo 1 – Trata da introdução do trabalho, aborda sobre os problemas e motivações para o projeto, bem como a justificativa e seus objetivos e a forma como o trabalho será organizado;

Capítulo 2 – Representa a parte destinada a fundamentação teórica do trabalho, descrevendo as características relativas ao material utilizado para o desenvolvimento do projeto;

Capítulo 3 – Destinado ao desenvolvimento, do qual constará a forma que será desenvolvido o presente trabalho;

Capítulo 4 – Apresenta os recursos de Hardware e Software utilizados no projeto.

Capítulo 5 – Demonstra de forma detalhada sobre o desenvolvimento do projeto de Software e o projeto de Hardware;

Capítulo 6 – Parte destinada aos testes e procedimentos de validação;

Capítulo 7 – Apresenta a conclusão sobre o projeto, bem como suas contribuições e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são apresentados os conceitos elementares das áreas compreendidas no desenvolvimento do presente trabalho.

Na Seção 2.1 são introduzidos alguns dos princípios da geoinformática e sua importância para sistemas de rastreamento. Em seguida, na Seção 2.2, são mostrados conceitos do funcionamento da rede GSM. A Seção 2.3 será destacado os conceitos no que se refere a plataforma de desenvolvimento que será utilizada. Na Seção 2.4 são apresentados os conceitos sobre os recursos eletrônicos que são controlados no veículo e, por fim, na Seção 2.5 são mostradas algumas informações sobre o equipamento de hardware que foi utilizado no presente projeto.

2.1 Geoinformática

O geoprocessamento pode ser entendido como o campo responsável pelo processamento informatizado de dados georreferenciados. Para ser considerado um dado georreferenciado, a imagem, mapa ou objeto mapeado, necessita ter suas coordenadas (longitude, latitude e altitude) conhecidas em um sistema de referência. A informação geográfica, quando possui uma referência que permite sua localização, pode ser chamada de geoinformação (COSTA, 2012).

Segundo Ehlers (2008), a Geoinformática é a arte, ciência ou tecnologia que lida com a aquisição, armazenamento, produção, processamento, apresentação e disseminação da informação geográfica. A sua essência é fundamentada sobre as tecnologias de aquisição, análise e visualização de dados espaciais.

Os dispositivos móveis atuais possuem cada vez mais poder de processamento e armazenamento. Esta característica fez com que os equipamentos que antigamente eram utilizados separadamente (mp3 player, PDA, computador, câmera fotográfica, celular, acesso Internet e módulos receptores GPS) fossem centralizados em um único equipamento: os *smartphones* e *tablets*.

Devido a esta revolução, os sistemas de rastreamento receberam um forte impulso que popularizou a tecnologia, abrindo o horizonte para as criações de novos aplicativos que podem ser usados para descobrir a posição de, por exemplo, um automóvel, outro *smartphone*, sendo que este último pode estar em qualquer lugar (no bolso de uma pessoa, dentro de um veículo, na mochila de uma criança, etc.), obtendo sua posição em tempo real.

Para utilizar geoprocessamento no presente projeto, basta desenvolver um software que recupere as coordenadas geográficas obtidas pelo receptor de GPS, envie para um *smartphone* através de uma SMS para que este processe os dados. Estes dados podem ser armazenados localmente no dispositivo, ou compartilhados na Internet.

2.1.1 Coordenadas Geográficas

O sistema referencial de localização terrestre é baseado em valores angulares expressos em graus, minutos e segundos de latitude (paralelos) e em graus, minutos e segundos de longitude (meridianos), sendo que os paralelos correspondem às linhas imaginárias Leste-Oeste ao Equador e os meridianos as linhas imaginárias de Norte-Sul, passando pelos polos, correspondentes a interseção da superfície terrestre com planos hipotéticos contendo o eixo de rotação terrestre (ROQUE et al., 2006). Uma representação gráfica das linhas de latitude e longitude é apresentada na Figura 1.

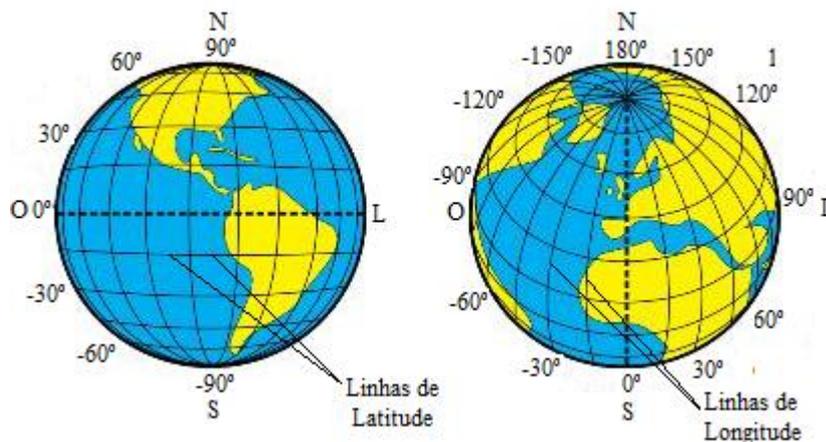


Figura 1 – O globo terrestre e suas linhas de latitude e longitude
 Fonte: Assis (2010).

Com as coordenadas geográficas torna-se possível a definição de qualquer ponto no globo terrestre. São linhas imaginárias verticais (longitude) e horizontais (latitude) que dividem a Terra em grandezas de graus, minutos e segundos. Sua interseção forma um ponto único em um plano bidimensional no globo terrestre.

A latitude é a coordenada geográfica de um ponto em relação ao Equador e é medida em graus, podendo variar entre 0° e 90° para Norte ou para Sul. Essas informações podem ser formadas por frações, graus, minutos e os segundos (que variam entre 0 e 59). “Por exemplo,

Pato branco está à latitude de $26^{\circ} 13' 44''$ S (26 graus, 13 minutos e 44 segundos Sul) (CONSTANTINI, 2009).

A longitude é a coordenada geográfica em relação ao meridiano de Greenwich medida ao longo da linha do Equador. Assim como a latitude, a longitude é medida em graus, minutos e segundos podendo este ser Leste ou Oeste. “Por exemplo, Pato Branco está à longitude de $52^{\circ} 40' 15''$ W (52 graus, 40 minutos e 15 segundos Oeste) (CONSTANTINI, 2009).

Para localidades situadas no oeste, é empregado o sinal de negativo (-) a frente do valor. Em alguns casos, autores empregam as letras "E" e "W" para indicações de Leste e Oeste, respectivamente. O casamento de valores de latitude e longitude obtém como resultado preciso um ponto em qualquer posição do globo terrestre (ASSIS, 2010).

2.1.2 GPS

Com o seu projeto iniciado em 1973, o sistema denominado *Navigation Satellite Timing and Ranging Global Positioning System* (Navstar GPS) mais conhecido como GPS, teve sua concepção por meio de fundos do Departamento de Defesa dos Estados Unidos para fornecer a posição instantânea e a velocidade de um ponto sobre a superfície terrestre, ou próximo a ela. Seu funcionamento é considerado pleno desde 1995. Atualmente, o GPS é utilizado por diversos segmentos na sociedade e seu uso está cada dia mais difundido (ASSIS, 2010).

O GPS é um sistema de navegação baseado em um conjunto de 27 satélites, sendo 24 deles operacionais e o restante são reservas, caso algum dos que estejam em operação venha a falhar, que tem órbitas com raio aproximado de 20 mil quilômetros e que completam aproximadamente duas voltas ao redor da Terra por dia. As órbitas são arranjadas de modo a possibilitar que, em um ponto qualquer da superfície terrestre, pelo menos, quatro satélites estejam visíveis, ou seja, acima da linha do horizonte (HASEGAWA et al., 1999).

Cada satélite transmite continuamente um sinal de rádio que inclui sua identificação, informações de sua órbita e o instante de tempo (horário) em que a transmissão foi feita, com precisão de um bilionésimo de segundo. Para que o sistema possuísse tal precisão, requereu-se que os satélites fossem equipados com relógios atômicos e que a sincronização dos seus horários levasse em conta efeitos relativísticos. Quando o sinal chega a um receptor GPS, este, com base nas informações do sinal, calcula a que distância do satélite ele se encontra. É necessário receber o sinal de no mínimo três satélites para que o receptor determine a longitude e a latitude (HASEGAWA et al., 1999).

Tecnicamente, cada satélite GPS contém um par de relógios atômicos com precisão de nano segundos, que constantemente enviam sinal para a superfície por meio de ondas de rádio, que viajam a velocidade da luz. Assim os receptores conseguem saber quanto o sinal viajou, sabendo quanto tempo o mesmo demorou para chegar. É este tempo, que permite com que o sistema aponte com precisão de metros uma determinada posição na Terra.

O trabalho do receptor GPS consiste em localizar pelo menos três satélites, determinar a distância para cada um deles e a partir dessas informações calcular sua própria posição. Essa operação é baseada no princípio matemático chamado trilateração (ARVUS, 2014, p. 1).

O princípio da trilateração em duas dimensões pode ser exemplificado com a seguinte analogia: para, por exemplo, determinar a localização representada pela Figura 2 (CANTO, 2010).

Se for conhecido apenas que este local está a 270 metros da estátua do centro da praça e a 200 metros do mastro da bandeira da escola, isto só permite concluir que pode ser o ponto A ou o ponto B. Mas, se for conhecido também, a distância a um terceiro ponto de referência, então a localização será inequívoca. Assim, se for conhecido que o local também está a 100 metros do poste ao lado da banca de jornal, então sua localização só pode ser o ponto A. Similarmente, quando um receptor GPS está informado da distância dos três satélites, pode determinar sua latitude e longitude. Nos aparelhos que fornecem velocidade e direção do movimento, estas são determinadas pela comparação de sucessivas posições ao longo do tempo. Além disso, se o receptor receber o sinal de um quarto satélite, é possível determinar, além da latitude e longitude, a altitude local (CANTO, 2010, p. 1).



Figura 2 – Analogia para determinação de coordenadas usando sinal GPS em 2D
Fonte: Canto (2010, p. 1).

O sistema GPS foi concebido para dar uma informação completa sobre a posição (latitude, longitude, altura e tempo). O conceito em duas dimensões também vale para sistemas tridimensionais, tomando agora, ao invés de círculos, esferas com raios definidos.

O processo de medição é um processo cíclico e elaborado. Em um determinado instante o satélite transmite um sinal digital, chamado PRC (*Pseudo Random Code*). No mesmo instante o receptor começa a gerar o mesmo código. Quando o sinal do satélite chega ao receptor haverá uma defasagem com o sinal gerado pelo receptor, a qual é o tempo de trânsito do sinal, logo o receptor multiplica o tempo pela velocidade da luz e determina a distância até o satélite. Porém, para executar esta medição ambos os relógios devem estar sincronizados com precisão de nano segundos. Para isso seriam necessários relógios atômicos em todos os satélites e ainda em todos os receptores, o que se tornaria inviável visto o custo de relógios atômicos (ARVUS, 2014, p. 1).

O GPS possui uma solução eficiente para este problema na qual apenas os satélites têm relógios atômicos e os receptores usam um cristal de quartzo comum, que não tem a mesma precisão dos relógios atômicos, mas é constantemente reiniciado em sincronia com os relógios dos satélites. Quando se mede a distância até quatro satélites, pode-se desenhar quatro esferas que se interceptam em um ponto. Três esferas interceptam-se mesmo se as medições não forem corretas. Porém, quatro esferas não se encontraram em um ponto se a medição for incorreta (ARVUS, 2014, p. 1).

Deve-se assumir também que certos fatores atmosféricos e outras fontes de erros, como lugares com grandes edifícios, podem afetar a precisão de um receptor GPS (GARMIN, 2014). Estes erros são resolvidos utilizando o GPS Diferencial (DGPS), que consiste na utilização de uma estação receptora fixa que recebe os sinais dos satélites e, como é um ponto fixo, pode calcular o erro que está sendo enviado pelos satélites e esta correção é enviada para todos os receptores GPS próximos que contemplam a tecnologia DGPS (ARVUS, 2014).

O conjunto de satélites GPS, mantido pelo governo americano, tornou-se operante em 1995. Na época, o sinal era codificado e, para os civis, permitia a precisão na ordem de 45 metros. A codificação foi removida em 2000 e, a partir de então, houve grande disseminação do uso de aparelhos GPS (HASEGAWA et al., 1999).

O Quadro 1 apresenta as vantagens e desvantagens mais significativas de se utilizar a tecnologia GPS.

Vantagens	Desvantagens
Abrangência mundial	Não funciona em ambientes fechados (túneis, garagens, etc.)
Adotado em 80% dos sistemas que utilizam GNSS ¹ .	Os sinais dos satélites podem ser obstruídos por pontes, viadutos, edifícios e matas.
Precisão de metros.	A geometria desfavorável dos satélites pode diminuir a precisão do posicionamento.
Não há custo de implantação e operação	O multicaminhamento, ou seja, a reflexão do sinal em algum objeto (como um prédio) interfere na precisão das coordenadas.
O preço dos equipamentos receptores vem diminuindo	Os receptores, ao serem ligados ou após perderem a comunicação com o satélite, levam certo tempo para reiniciar a aquisição de dados.

Quadro 1 – Vantagens e desvantagens da tecnologia GPS

Fonte: Assis (2010).

Para as transmissões dos dados referentes a posição adquiridas através do GPS é utilizado o protocolo NMEA (*National Marine Electronics Association*), que é um conjunto de especificações de dados para comunicação de dispositivo eletrônicos utilizado nos receptores GPS.

2.1.3 Protocolo NMEA 0183

O NMEA 0183, ou simplesmente NMEA, é uma norma que foi criada pela Marinha Americana com o propósito de organizar e padronizar as informações recebidas dos satélites. O nome faz referência ao seu órgão de criação, *National Marine Electronics Association*, que nasceu como uma associação de fabricantes de produtos eletrônicos norte-americanos, mas que hoje tem membros espalhados por todo o mundo (FERREIRA, 2010).

O protocolo de comunicação NMEA baseia-se no envio unidirecional de dados entre o GPS e outro dispositivo, através de sentenças no formato ASCII com códigos específicos.

Com a unificação da comunicação conseguida pelo padrão NMEA, facilitou-se muito o desenvolvimento e integração de equipamentos eletrônicos no campo da navegação. Esse padrão é usado pela maioria dos receptores GPS (FERREIRA, 2010).

Segundo Ferreira (2010), os dados são enviados através de mensagens de texto em que cada linha identifica certo tipo de dados do GPS. Alguns dos tipos de dados dessas linhas são:

¹ GNSS (*Global Navigation Satellite System*) é o nome genérico usado para denominar os sistemas de navegação por satélite que permitem a denominação de posicionamento geoespacial com cobertura global.

- GPGGA – Global Positioning System Fixed Data.
- GPGLL – Geographic position, Latitude and Longitude.
- GPGSA – Satellite status.
- GPGSV – Satellites in view.
- GPRMC – Recommended Minimum Specific Data.
- GPVTG – Track made good and ground speed.
- GPZDA – Data and Time.

A ideia do padrão é fornecer informações em linha de forma estruturada para assim normalizar a configuração e o controle dos mecanismos de dados do receptor GPS. O exemplo do Quadro 2 descreve todos os parâmetros da resposta NMEA \$GPRMC, que foi o utilizado no presente projeto. Dentre as informações que são transmitidas pelo módulo GPS, destaca-se os dados de posicionamento (latitude, longitude e altitude) e a hora do satélite.

Exemplo:

\$GPRMC,161229.487,A,3723.2475,N,12158.3416,W,0.13,309.62,120598,*,*10

Nome	Exemplo	Unidade	Descrição
Message ID	\$GPRMC		RMC protocol header
UTC Time	161229.487		hhmmss.sss
Status	A		A = data valid or V = data not valid
Nome	Exemplo	Unidade	Descrição
Latitude	3723.2475		ddmm.mmmm
N/S Indicator	N		N = North or S = South
Longitude	12158.3416		dddmm.mmmm
E/W Indicator	W		E = East or W = West
Speed Over Ground	0.13	Knots	
Course Over Ground	309.62	Degrees	
Date	120598		ddmmyy
Magnetic Variation		Degrees	E = East or W = West
Checksum	*10		

Quadro 2 – Formato dos dados do protocolo GPRMC

Fonte: NMEA (2008).

2.2 A rede de comunicação das operadoras de telefonia móvel

A ideia da comunicação instantânea, independente da distância, é um dos sonhos mais antigos do homem e esteve sempre ligado ao progresso da eletrônica, das tecnologias e dos recursos disponíveis. Inicialmente, as telecomunicações surgiram como sistemas fixos de envio de mensagens como telégrafo, inventado em 1838 pelo norte americano Samuel Morse, que enviava sinais elétricos codificados através de cabos metálicos. Muitas foram as tentativas de aperfeiçoar o telégrafo até que Alexander Graham Bell conseguiu transmitir sons através dos fios pelo telégrafo falante, que mais tarde seria batizado de telefone (SILVA, 2010).

No decorrer da história da telefonia celular, um dos maiores desafios é absorver as tecnologias e as novidades que se apresentam em nosso cotidiano. Devido a isso, houve algumas evoluções nas gerações desses dispositivos. Os serviços oferecidos na primeira geração eram serviços muito simples, basicamente serviços de voz, que permitiam uma pessoa originar e receber chamadas com mobilidade. E, após gerações de tecnologia de celulares, foram desenvolvidos as redes digitais, que oferecem recursos multimídia para transmissão de dados. Estes primeiros equipamentos funcionavam com recursos a redes 1G, redes móveis de primeira geração. Esta era uma tecnologia analógica que apenas permitia a comunicação por voz, não suportando a transmissão de dados. Já no final da década de 80, o sistema analógico evoluiu para um sistema digital, que acrescentou as comunicações móveis a transmissão de dados e após alguns anos surgiram as redes digitais, que oferecem recursos multimídia para a transmissão de dados, o que é possibilitado pelas redes 2G e 3G. Este último possibilita a navegação pela internet no dispositivo móvel em alta velocidade (SILVA, 2010).

Em 1992, surgiu a primeira comunicação utilizando mensagens curtas de texto, este chamado de SMS (*Short Message Service*), este já era composto por um protocolo simples, que estipulava algumas regras, como o formato dos campos transferidos e também o limite de caracteres (PERALTA, 2012).

A rede GSM utilizadas pela maioria das operadoras brasileiras é um padrão internacional que planejou unificar os padrões de telefonia celular. Este acabou sendo o primeiro padrão digital a ser utilizado comercialmente e que se difundiu pelo mundo fornecendo aos dispositivos móveis maior mobilidade nos serviços de voz e dados, com boa qualidade e custos menores (COSTA, 2012).

Para o presente projeto, adotou-se um *smartphone* que possui tecnologia GSM, onde por intermédio desta tecnologia é possível a transmissão das informações do veículo por

intermédio do envio de SMS o qual o aplicativo no *smartphone* processa estas informações e faz o uso da mesma. Este último também envia comando para o microcontrolador através do uso de um módulo GSM que recebe a SMS com a ação a ser tratada pelo microcontrolador.

2.3 Plataforma de desenvolvimento para dispositivos móveis Android

O Android é uma plataforma para desenvolvimento e execução de programas para dispositivos móveis, robusta e de fácil utilização. Esta plataforma é composta por um sistema operacional e um ambiente de desenvolvimento de software.

Sistema operacional é um conjunto de programas que gerenciam todas as tarefas de um dispositivo, fornecendo ao usuário uma interface visual para interagir com o sistema eletrônico do dispositivo sem a necessidade do mesmo saber o que acontece dentro dele (TANENBAUM, 2009).

Para o desenvolvimento de aplicativos para um sistema operacional, é necessário um ambiente de desenvolvimento de software, composta por uma linguagem de programação, que permita criar aplicativos que interfaceiam os comandos realizados pelo usuário e os recursos do sistema operacional.

Quando os dispositivos móveis começaram a agregar funcionalidades surgiu a necessidade da criação de sistema para gerenciá-las. Com isso, os dispositivos móveis como *smartphones* e *tablets* começaram a se difundir.

Os dispositivos móveis com suporte ao sistema operacional Android foram lançados em 2008. O primeiro *smartphone* Android foi o HTC Dream – Figura 3, lançado em outubro do mesmo ano. Apesar de lançado em 2008, a plataforma Android teve seu início no ano de 2003, sendo concebido com base no sistema operacional Linux (G1, 2008).



Figura 3 – HTC T-Mobile, primeiro aparelho com Android

Fonte: G1 (2008).

Foi em 2003 que a empresa Android Inc. foi fundada por Andy Rubin. A empresa tinha como objetivo o desenvolvimento de um sistema operacional para dispositivos móveis, dada como necessidade as limitações existentes nos aparelhos celulares da época (DANA, 2011).

Em 2005, o projeto da empresa acabou chamando a atenção da Google, que já estava interessada em expandir sua influência para o mundo de tecnologia móvel, fazendo com que a mesma comprasse a Android Inc. e utilizasse seu projeto no mercado de dispositivos móveis (DANA, 2011).

Uma das principais vantagens dos *smartphone* com a tecnologia Android é a total integração com os serviços do Google, por exemplo, o Gmail, Gtalk, Google Docs e Google Maps. Outra vantagem é a fácil integração com recursos nativos do aparelho, como manipulação dos dados existentes na agenda do aparelho, utilização de recursos como Bluetooth e GPS, já que o ambiente de desenvolvimento Android permite a utilização da maioria dos recursos existentes no sistema operacional Android (ZOOM, 2014).

2.4 Recursos eletrônicos presentes nos veículos

Com o passar dos anos, os engenheiros e projetistas de automóveis começaram a substituir seus controles mecânicos por dispositivos eletrônicos. Hoje, é muito difícil encontrar um carro com carburador, por exemplo, já que este sistema foi substituído pela injeção eletrônica. Outros recursos começaram a ser modernizados e estão cada vez mais presentes nos veículos de hoje, como os vidros elétricos, trava de portas eletrônicas, computador de bordo, entre muitos outros recursos tecnológicos.

A trava elétrica consiste em um pequeno motor que é instalado próximo a fechadura da porta do veículo. Possui um interruptor e sensores elétricos que, quando acionados, trancam ou destrancam a porta. No caso dos vidros elétricos, estes são constituídos de um motor elétrico e um conjunto de botões responsáveis pelo acionamento de subida ou descida dos vidros.

Também vale destacar que os automóveis sempre foram alvo natural para os ladrões. Devido a isso, as pessoas começaram a investir em sistemas eletrônicos de alarmes com o objetivo de prevenir que seus automóveis sejam roubados. Esses equipamentos são equipados com sensores eletrônicos, sirenes e sistema de ativação remota.

Um alarme de um automóvel é basicamente um conjunto de sensores unidos a algum tipo de sirene. Os sistemas de alarme podem ser compostos por:

- Uma série de sensores que podem incluir interruptores, sensores de pressão e detectores de movimento;
- Uma sirene, responsável por emitir sons;
- Um receptor de rádio para permitir controle sem fio a partir de um outro equipamento de mão;
- Uma unidade de controle responsável por monitorar os sensores e se houver alguma mudança soar o alarme.

Alguns sistemas de alarme em veículos integram a função de travar e destravar as portas quando o mesmo é ativado ou desativado respectivamente, além de fechar as janelas caso o veículo possua vidros elétricos.

O controle das travas elétricas e dos vidros elétricos por sua vez também podem ser controlados separadamente do alarme. Ambos consistem em um sistema eletromecânico agregado a porta e podem ser acionado remotamente a partir de um controle, por exemplo.

2.5 Microcontroladores

Segundo Denardin (2013), um microcontrolador é um sistema computacional completo, no qual está incluída uma CPU (*Central Processor Unit*); memória de dados e programa, EEPROM (*Electrically-Erasable Programmable Read-Only Memory*) ou memória *Flash* (uma variação das EEPROM) para armazenamento permanente de dados; um sistema de *clock*; portas de I/O (*Input/Output*); além de outros possíveis periféricos, tais como, módulos de temporização e conversores A/D (Analogico/Digital) entre outros, integrados em um mesmo componente. As partes integrantes de qualquer computador, e que também estão presentes, em menor escala, nos microcontroladores são:

- Unidade Central de Processamento (CPU);
- Sistema de *clock* para dar sequência às atividades da CPU;
- Memória para armazenamento de instruções e para manipulação de dados;
- Entradas para interiorizar na CPU informações do mundo externo;
- Saídas para exteriorizar informações processadas pela CPU para o mundo externo;
- Programa (*firmware*) para definir um objetivo ao sistema.

A Figura 4 apresenta um diagrama de blocos genérico de um microcontrolador.

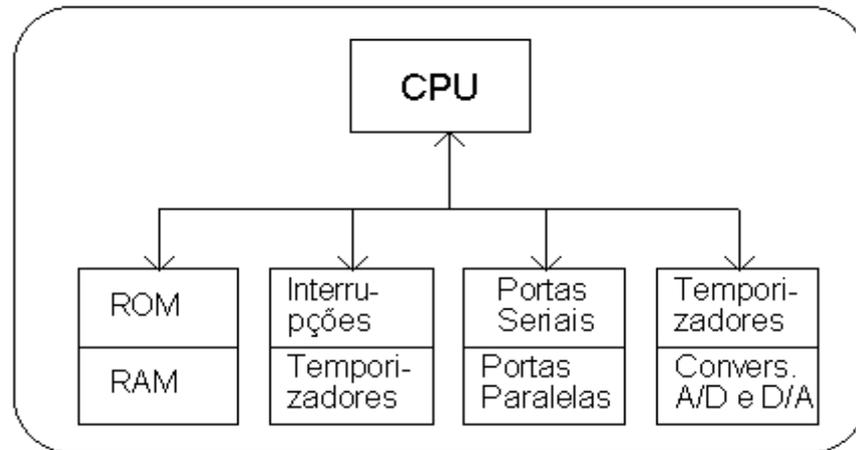


Figura 4 – Diagrama de blocos de um microcontrolador

Fonte: Zelenovsky, Mendonça (2014).

Os microcontroladores normalmente são classificados em famílias, dependendo da aplicação a que se destinam. A partir da aplicação que a família de microcontroladores se destina, um conjunto de periféricos específicos é escolhido e integrado a um determinado microprocessador. Estes microprocessadores normalmente operam com barramentos de 8, 16 ou 32 bits, e apresentam arquiteturas RISC (*Reduced Instruction Set Computer*) ou CISC (*Complex Instruction Set Computer*) (DENARDIN, 2013).

Os microcontroladores podem ser utilizados em automação e controle de produtos e periféricos, como sistemas de controle de motores automotivos, controles remotos, máquinas de escritório e residenciais, brinquedos, sistemas de supervisão, entre outros.

2.6 Trabalhos Relacionados

Esta seção tem como objetivo apresentar os produtos existentes no mercado que possuem uma ideia similar ao proposto pelo presente trabalho.

2.6.1 Sascar APP

A Sascar (2013) é uma empresa especializada em soluções para gestão das operações de transporte que visa prover soluções intuitivas e fáceis de usar aos gestores de transporte, em tempo real, via *web* ou através de *smartphones*, permitindo o monitoramento dos veículos e acompanhamento do consumo de combustível, quilometragem, entre outras ações, reduzindo os custos operacionais e aumentando a produtividade.

Com esse objetivo, a empresa criou o Sascar APP, representado na Figura 5, que é um aplicativo para *smartphone* que deixa o proprietário do veículo conectado ao mesmo. Com esse aplicativo é possível monitorar o veículo em tempo real, rastreando as últimas posições. Caso o veículo saia de um perímetro determinado, é enviado um alerta para o celular, criando uma rota do celular até o local onde o veículo se encontra. Este é um aplicativo gratuito e está disponível para iPhone e Android.



Figura 5 – Sascar APP
Fonte: Sascar (2013).

2.6.2 Netraster APP

A Netraster (2013) é uma empresa que atua na segurança eletrônica e monitoramento de veículos. Para maior comodidade a empresa criou um aplicativo de controle remoto via celular, representado na Figura 6, que permite controlar algumas funcionalidades do veículo através da tecnologia Bluetooth.

O aplicativo consiste em: acionar o alarme ou buzina, ligar e desligar a ignição do motor, acender e apagar os faróis, abrir e fechar os vidros, travar e destravar as portas. Tudo isso é feito através da tecnologia Bluetooth.



Figura 6 – Netraster APP
 Fonte: Netraster (2013).

2.6.3 My BMW Remote

A BMW é uma montadora de automóveis alemã, que está sempre investindo em novas criações que elevem seu sucesso a curto e em longo prazo. Visando esta meta, a empresa lançou um aplicativo em Android que controla algumas funcionalidades do veículo, o *My BMW Remote* (BMW, 2010), representado na Figura 7, que transforma o aparelho celular em um controle remoto do automóvel.

O aplicativo, que já havia sido lançado para a plataforma IOS, que é o sistema operacional móvel da empresa Apple (Apple Brasil, 2013), permite que o condutor trave e destrave as portas do veículo, controle a temperatura interna do veículo, use a buzina ou encontre o automóvel via GPS e com o auxílio dos piscas. No caso da localização, o GPS traça a rota de onde o veículo está, tudo isso através da Internet. Além disso, o programa funciona apenas nos modelos que possuem o *BMW Assist Safety Plan*. O aplicativo é fornecido pela empresa gratuitamente para plataforma Android e IOS (PIMENTEL, 2012).



Figura 7 – My BMW Remote
 Fonte: BMW (2013).

2.6.4 BL Free 906

A LocatorOne é uma empresa brasileira que cria e comercializa bloqueadores e rastreadores veiculares. Com este fim, a empresa criou o bloqueador BL Free 906, representado na Figura 8, que avisa o proprietário do veículo através de uma ligação para o celular, caso o mesmo tenha sido violado, o proprietário pode então, cortar o funcionamento do motor através do próprio aparelho digitando uma senha (VANDRÉ, 2007).



Figura 8 – Bloqueador BL Free 906
 Fonte: Vandr  (2007).

2.6.5 Sistema de alarme integrando GPS e celular

Este é um trabalho acadêmico realizado pelo acadêmico Daniel Alves Ferreira, em 2010, pelo Centro Universitário de Brasília (UniCEUB). Trata-se de um sistema de alerta que avisa ao proprietário de um veículo quando o mesmo entrar em movimento não autorizado, através de uma ligação para o número de telefone previamente armazenado no sistema. Se o movimento não autorizado acontecer, o proprietário será avisado para que possa tomar providências imediatamente após o ocorrido.

Quando o proprietário estacionar seu carro, o mesmo pressionará um botão que iniciará o sistema, realizando as funções de captura das coordenadas, análise das mesmas e realização da ligação telefônica. No modelo proposto, o alerta é realizado a partir de uma movimentação de 15 metros em função da precisão do GPS.

3 METODOLOGIA

Nos últimos anos, o uso dos *smartphones* sofreu uma grande expansão. Cerca de 14% da população brasileira já possui um *smartphone*, e dentre estes, 79% dos aparelhos possuem sistema operacional Android (ÉPOCA, 2012). Aproveitando a possibilidade que este cenário traz, novas aplicações vem surgindo, as quais utilizam tanto a rede de telefonia móvel para o processamento distribuído quanto recursos nativos dos *smartphones* (ALVEZ, 2011). Como exemplo desses sistemas pode-se citar: controle de alarme de automóveis, sistemas de rastreamento de veículos, sistemas de autenticação de usuários, entre outros.

Devido aos *smartphones* estarem cada vez mais incorporados no dia a dia das pessoas e possuírem cada vez mais funcionalidades, o presente projeto tem como objetivo o desenvolvimento de uma solução em Android para controlar algumas funcionalidades de um veículo, assim como recuperar dados do veículo. São características do projeto proposto:

- Rastrear a posição em que o veículo se encontra em determinado momento;
- Controle recursos do veículo, tais como alarme, travas elétricas e vidros elétricos das portas;

A Figura 9 retrata a ideia principal do sistema, no qual o *smartphone* irá enviar/receber informações sobre uma respectiva ação. O módulo GSM (SIM900) irá se comunicar com o microcontrolador, que por sua vez será responsável pela execução de determinada ação dentro do veículo e este também irá enviar as coordenadas (latitude e longitude) adquiridos do módulo GPS (NEO-6M) para o *smartphone*, para que este último possa tratar esses dados e apresentar em um mapa a localização do veículo.

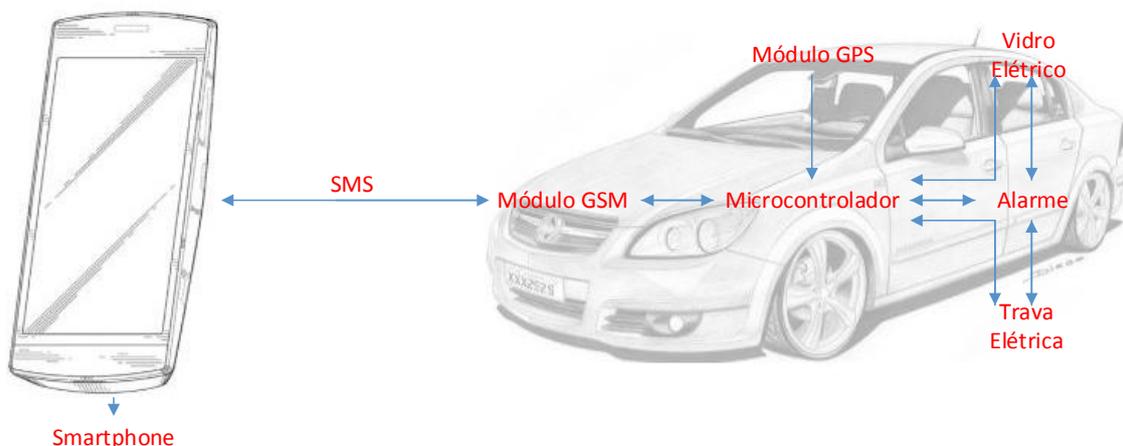


Figura 9 – Esquema básico de funcionamento do sistema

Fonte: Autoria própria.

O desenvolvimento do protótipo sugerido neste projeto foi dividindo em etapas de acordo com o modelo essencial de Pressman (2002). Essas etapas são:

- Requisitos: A definição dos requisitos foi realizada a partir da análise das necessidades de se automatizar alguns recursos do veículo, verificando o modo de utilização dos recursos e como estes são dispostos aos usuários;
- Análise: Os casos de uso foram definidos como forma de representar o problema para o qual seria proposta uma solução. Os casos de uso foram documentados utilizando um editor de textos e a ferramenta Astah Community para produzir os diagramas UML;
- Projeto: Os diagramas de classes, definidos com a linguagem UML, foram produzidos utilizando a ferramenta Astah Community;
- Desenvolvimento: O desenvolvimento foi realizado utilizando as tecnologias listadas na Seção 4.1.

O projeto pode ser separado em duas vias, sendo uma delas o controle do rastreamento do veículo e a outra o controle de algumas funcionalidades que serão especificadas.

Vale ressaltar que o sistema só funcionará onde existir a cobertura da rede da operadora e onde seja possível a aquisição dos dados referentes a posição do receptor GPS, logo, caso o usuário estiver em um local onde não existir a mesma, o sistema ficará inativo.

Levando em consideração esse aspecto, a ativação/desativação do alarme do veículo também deverá funcionar através de um controle remoto em posse do usuário.

3.1 Rastreamento do veículo

Para realizar o rastreamento do veículo, este estará equipado de um módulo GSM (SIM900) e um módulo GPS (NEO-6M), demonstrado na Figura 9. O módulo GPS é responsável por adquirir as informações relativas a posição (latitude e longitude) do veículo, a qual será transmitida pelo módulo GSM para o *smartphone* Android através de um SMS, este último irá tratar a informação recebida para que seja possível o usuário verificar em um mapa a atual posição do veículo.

O receptor GPS será responsável por obter as informações relativas ao posicionamento do veículo e através do Módulo GSM estas serão enviadas a cada 10 minutos (tempo pré-estabelecido pelo desenvolvedor) para o dispositivo móvel do usuário e estes dados serão armazenados em um banco de dados presente no mesmo.

O dispositivo móvel irá apresentar para o usuário uma lista das últimas posições do veículo, sendo que estas podem ser vistas em um mapa. Para isto, serão utilizados os recursos oferecidos pela Google Maps API.

3.2 Acionamento dos recursos do veículo

No que se refere ao controle dos recursos do veículo, o projeto proposto visa à ativação/desativação remota do alarme via um sinal emitido pelo *smartphone* usando para isso o envio de um SMS para o módulo GSM. Este sinal será recebido pelo módulo GSM, que por sua vez irá realizar a comunicação com o microcontrolador, o qual é responsável por realizar o acionamento, se caso o alarme for ativado, travar as portas e fechar os vidros, ou se caso o alarme for desativado, somente destravar as portas.

Além disso, o usuário também poderá controlar individualmente as travas elétricas e os vidros elétricos remotamente, caso o alarme do veículo esteja desativado.

4 RECURSOS DE HARDWARE E SOFTWARE

Nesta seção são apresentados os recursos de hardware e software a serem utilizados no desenvolvimento do projeto, sendo este formado por um microcontrolador Tiva TM4C123G, um kit Launch Pad utilizado para o desenvolvimento do projeto. A fim de simular o alarme veicular foi utilizado um sensor PIR (*Passive Infrared*) e um *speaker*. Um servo motor para simular a trava elétrica e por último um motor de DC para simular o vidro elétrico.

4.1 Recursos de hardware

Nesta seção são apresentados os recursos de hardware que serão utilizados no desenvolvimento do projeto.

4.1.1 Microcontrolador

Para este projeto, foi usado especificamente o Tiva TM4C123G LaunchPad Evaluation Kit, desenvolvido pela Texas Instruments®, que pode ser adquirido por um custo acessível e é uma plataforma para desenvolvimento em microcontroladores baseados na tecnologia ARM® Cortex™- M4F, também desenvolvidos pela Texas Instruments®.

O Tiva LaunchPad Evaluation Kit é exibido da Figura 10, algumas das principais características deste kit são (TEXAS INSTRUMENTS, 2013).

- Microcontrolador TM4C123G:
 - 32 bits com ponto flutuante;
 - Operação de até 80 MHz;
 - 256KB de memória Flash;
 - 32 KB de memória RAM;
 - Modo de hibernação;
- 0 – 43 GPIO;
- 2 ADCs (Conversores analógico/Digital) 1MSPS de 12 bits;
- Até 27 temporizadores;
- LED (*Diode Emitter de Luz*) RGB (*Red Green Blue*);
- Conectividade USB;
- Botão de *reset*;
- 2 *push-buttons*;

- Cristal oscilador principal de 16 MHz;
- Conectividade serial:
 - USB (*Universal Serial Bus*) 2.0;
 - 8 UART (*Universal Asynchronous Receiver/Transmitter*);
 - 4 I2C (*Inter-Integrated Circuit*);
 - 4 SSI / SPI (*Synchronous Serial Interface/Serial Peripheral Interface*);
- Entre outros;

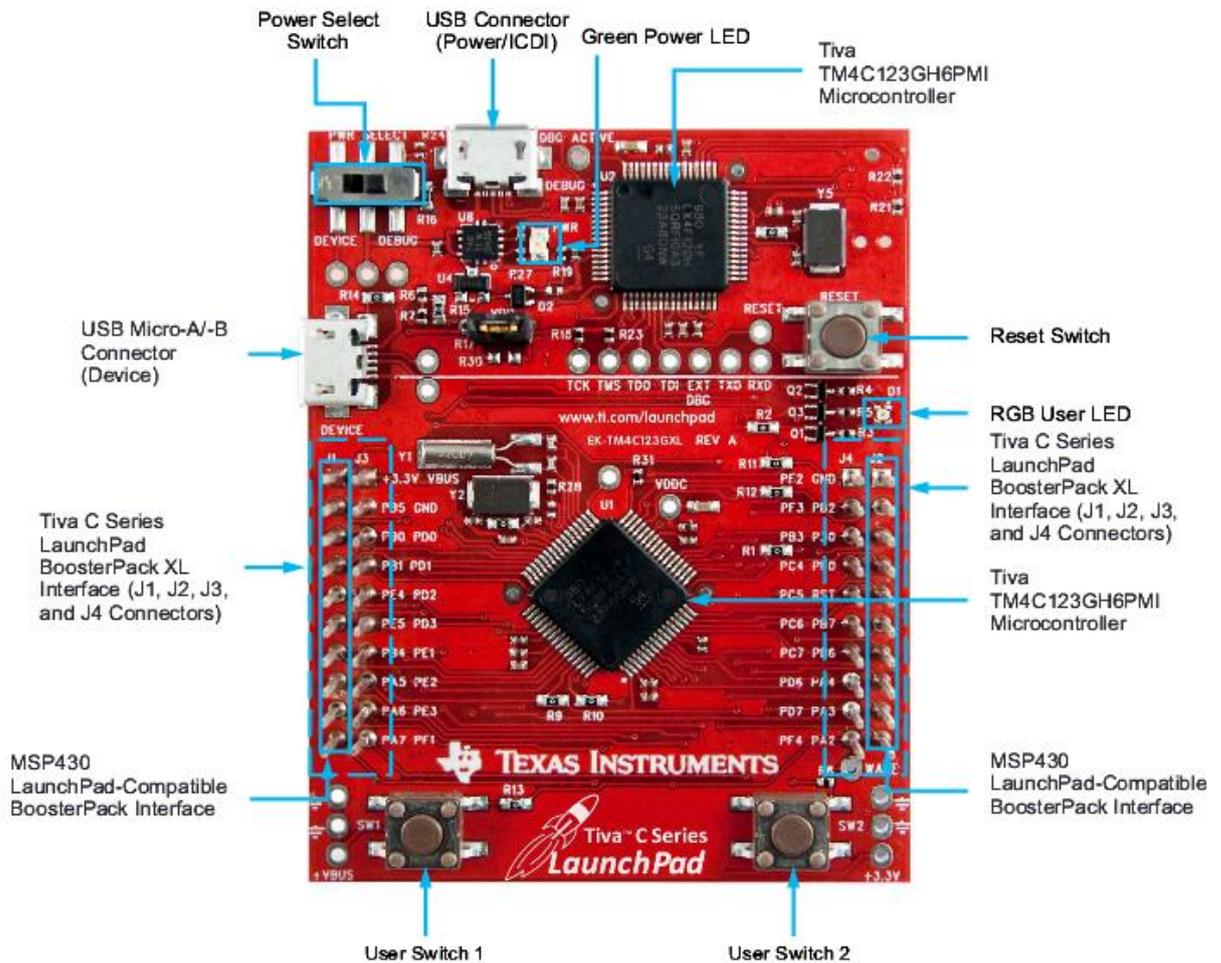


Figura 10 – Tiva LaunchPad
 Fonte: Texas Instruments (2013).

4.1.2 Módulo GSM/GPRS SIM900 S2-1040S

Shields são placas de circuitos contendo outros dispositivos, por exemplo, receptores GPS, *displays* de LCD, módulos de Ethernet, etc. que se ao conectar a um microcontrolador,

este obtém funcionalidade adicionais. Shields também estendem pinos até o topo de suas placas de circuito, para continuar tendo acesso a todos (SILVA, 2012).

A Figura 11 representa a *shield* GSM/GPRS baseado no módulo *quad-band* SIM900 com pilha TCP/IP embutida que pode ser usada em aplicações em que a transmissão via tecnologia GSM/GPRS é necessária, seja por transmissão de voz ou SMS. É controlado via comandos AT e possui um consumo reduzido de energia. Ideal para aplicações industriais de telemetria.



Figura 11 – SIM900 S2-1040S
Fonte: Autoria própria.

As principais características deste módulo são:

- Quad-band 850/900/1800/1900 MHz;
- GPRS mult-slot classe 10/8;
- Controle via comando AT (GSM 07.07, 07.05, mais os comandos AT da SIMCom);
- Baixo consumo de energia: 1,5mA;
- Voltagem de operação: 3,2V a 4,8V;
- Temperatura de operação: -40°C a 85°C;
- Suporte para ser ativado via software.

Os comandos AT, antigamente chamados de *Hayes Command Set*, implementam uma linguagem de programação que também é utilizada para operar alguns modelos de *modems* remotamente (SILVA, 2012).

Uma linha de comando AT pode conter um ou mais comandos, utilizando delimitadores para separar cada um. A linha possuirá o prefixo “AT” e o sufixo, ASCII de *Carriage Return*, o delimitador poderá ser um ponto e vírgula ou um espaço. O modem

emitirá uma mensagem, *Result Code*, no momento em que o comando é emitido, avisando assim ao terminal o resultado do comando requisitado (SILVA, 2012).

O módulo SIM900 será responsável por executar os serviços que tratam SMS por meio de comandos AT.

4.1.3 Módulo GPS NEO-6M

O NEO-6M é um receptor GPS de alta performance e de baixo custo, ideal para aplicações em dispositivos móveis. A Figura 12 apresenta este receptor GPS integrado com uma *shield*.



Figura 12 – Módulo GPS NEO-6M

Fonte: Autoria própria.

As principais características deste módulo são:

- Possui uma interface para cartão SD (que pode ser armazenado os dados referentes a localização geográfica);
- Alta sensibilidade;
- Taxa de atualização de até 5Hz;
- Utiliza o protocolo NMEA 0183 como formato de dado de saída;
- Precisão de aproximadamente 2,5 metros
- Opera entre 2,7V a 3,6V;
- Possui interface UART, USB e SPI;
- Precisão do sinal: 30ns;
- Temperatura de operação: -40°C a 85°C;

Além destas características este receptor GPS conta com a tecnologia *Assisted* GPS (A-GPS), neste caso, a conexão não é feita diretamente entre o satélite e o receptor GPS, a primeira conexão é feita com antenas de telefonia móvel que possui a localização dos satélites armazenada. Assim, as informações são transmitidas em uma velocidade maior. O A-GPS também conta com antenas para manter o sinal estável em regiões com prédios altos para que o receptor funcione corretamente.

4.1.4 Alarme veicular

O sensor PIR é um sensor de movimento por calor (infravermelho). Assim que ocorrer uma mudança no infravermelho em um determinado lugar, esta mudança será captada pelo sensor. Portanto, se uma pessoa, por exemplo, entrar na área de alcance do sensor, a mesma será detectada devido a energia infravermelha irradiada pelo corpo da pessoa. Devido a esta característica, este tipo de sensor é muito utilizado em sistemas de alarme, controle de pedestres, iluminação automática, entre outros.

No presente projeto é utilizado o sensor PIR HC-SR501, mostrado na Figura 13.

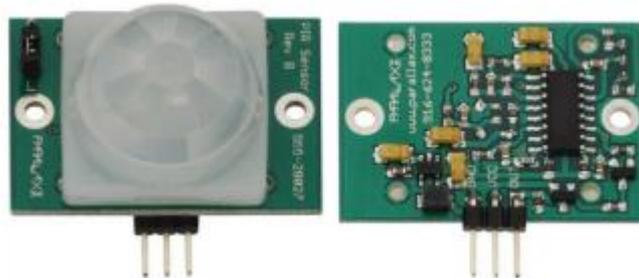


Figura 13 – Sensor PIR HC-SR501
Fonte: PIR Sensor (2012).

O sensor possui três pinos, sendo eles (PIR SENSOR, 2012):

- Alimentação: de 4,5V a 20V;
- GND: terra;
- Dados: usado para fazer o envio de informações para o microcontrolador, no caso, somente indica se foi detectado algum movimento ou não (nível alto ou nível baixo).

Este sensor tem um alcance de até 7 metros com ângulo de detecção de aproximadamente 120 graus. Possui tensão de saída de 3,3V e corrente de repouso de 50 μ A. A temperatura de operação varia de -15° até 75° (PIR SENSOR, 2012).

O *speaker* (Figura 14) é um dispositivo usado para reproduzir sons. O mesmo converte a energia elétrica em energia acústica. Para isto, inicialmente ele converte a energia elétrica em vibrações mecânicas para então transformar a energia mecânica em acústica. Este equipamento foi usado para emitir o um *beep* caso o protótipo estiver em uma condição de risco.



Figura 14 – Speaker
Fonte: Autoria Própria.

4.1.5 Servo motor

O servo motor é um atuador rotativo com alta precisão de controle para posicionamento angular. É composto por um motor acoplado a um sensor de posição para feedback. Além disso, o servo motor não possui rotação contínua, ou seja, trabalha dentro de um limite de ângulos específicos. Para completar o sistema é preciso um drive que utiliza o sinal de feedback do sensor para controlar a posição angular do motor de forma precisa. Isto é chamado de controle a malha fechada.

O servo motor utilizado no presente projeto é o Tower Pro SG-5010, este representado pela Figura 15, que apresenta as seguintes características:

- Voltagem de operação: 4,8V a 6,0V;
- Modulação: Analógica;
- Velocidade: 0,17 seg/60° (4,8V) e 0,14 seg/60° (6,0V);
- Torque máximo: 8,0 kg.cm (4,8V) e 11,0 kg.cm (6,0V);
- Temperatura de operação: -20°C a 60°C;
- Faixa de rotação: 180°.



Figura 15 – Servo Motor Tower Pro SG-5010
Fonte: Autoria Própria.

4.1.6 Motor DC

O motor DC se baseia no princípio de atração e repulsão dos polos magnéticos. Basicamente, uma espira é mergulhada em um fluxo magnético e quando esta espira sobre uma indução de tensão elétrica é provocada uma rotação da espira em seu próprio eixo. Logo após esta rotação o inversor localizado nas extremidades da espira alterna a direção da tensão elétrica que conseqüentemente repete a ação de atração do campo eletromagnético da espira com o campo magnético do ímã permanente e assim o processo se repete continuamente gerando uma rotação continua no eixo da espira.

A indução de tensão elétrica sobre um condutor (a espira) gera um campo eletromagnético ao seu redor. Os polos deste campo eletromagnético sofrem atração e repulsão pelo polos do campo magnético em que se encontra submerso.

A direção rotacional do eixo é definida pela direção do fluxo magnético (do ímã permanente) e pela direção da corrente da tensão elétrica.

Para o presente projeto optou-se utilizar um motor DC de 12V de 80RPM com caixa de redução (Figura 16) que apresenta as seguintes características:

- Rotação: 80RPM;
- Corrente: 0,05A;
- Tensão de alimentação: 12V DC;
- Caixa de redução;
- Torque: 3kg.cm;
- Comprimento do eixo: 9,5mm;
- Diâmetro do eixo: 4mm.



Figura 16 – Motor DC
Fonte: Autorial própria.

Para fazer um motor DC rodar em ambas as direções deve-se usar transistores e diodos de proteção. O L293B (Figura 17) é uma ponte H dupla, isto é, permite ligar dois motores e fazer com que estes rodem em ambas as direções. Com este circuito integrado é possível controlar motores de até 36V de alimentação, porém aguenta corrente constante de 600mA e corrente de pico de 1,2A.

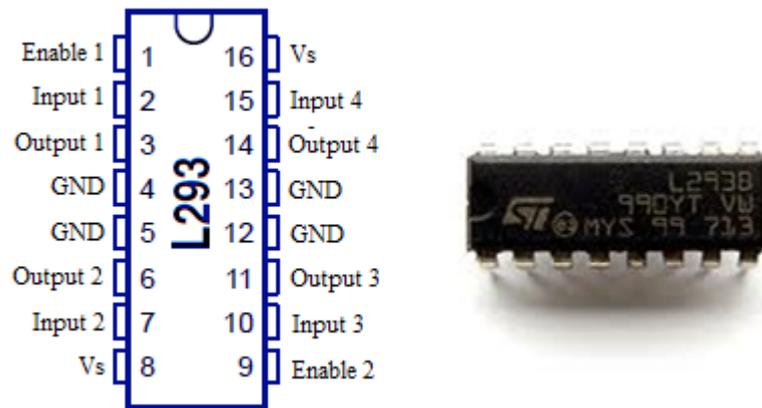


Figura 17 – L293B
Fonte: Adaptado Datasheet L293B (1993).

4.2 Recursos de software

Nesta seção são apresentados os recursos de software utilizados no desenvolvimento do projeto, como linguagens de programação e IDEs.

A Figura 18 apresenta de uma forma clara onde cada recurso de software foi utilizado no desenvolvimento do presente projeto.

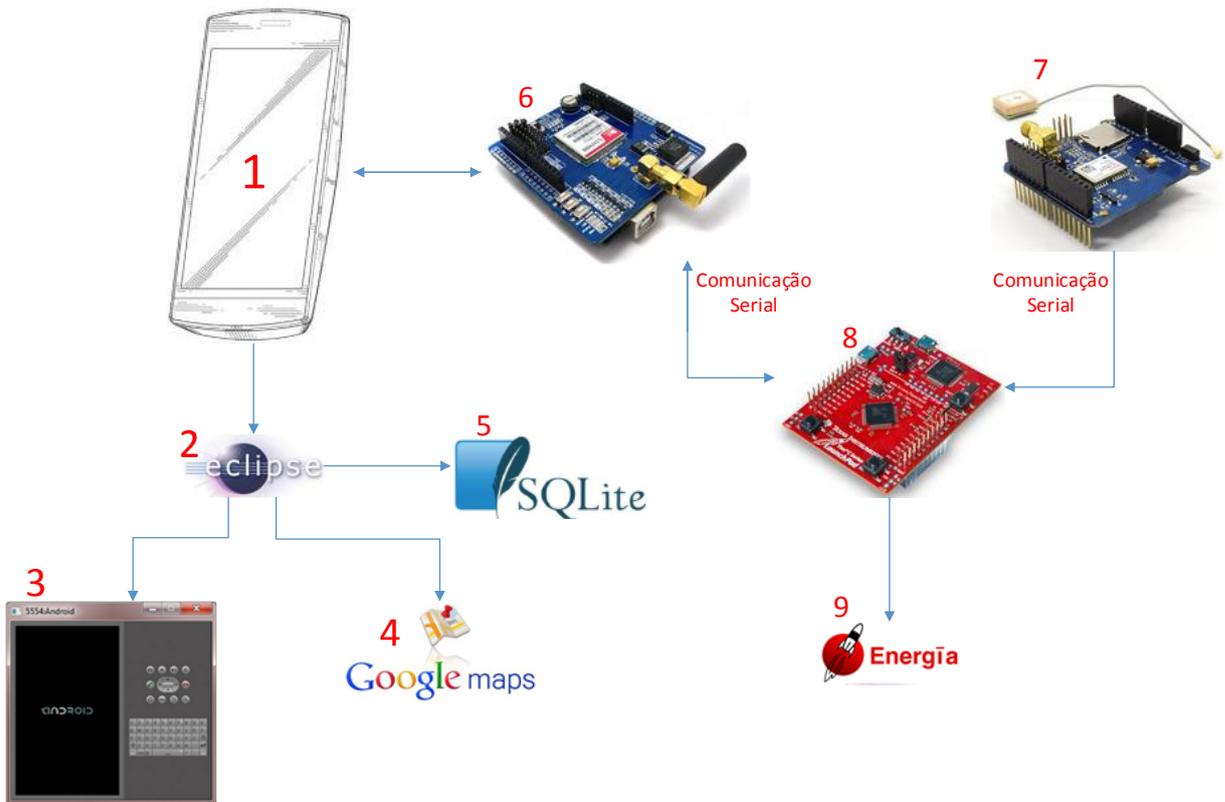


Figura 18 – Recursos de software presentes no projeto

Fonte: Autoria própria.

Na sequência a identificação de cada item da Figura 18:

1. *Smartphone* Android;
2. IDE Eclipse;
3. *Android Virtual Device* (AVD);
4. Google Maps API;
5. SQLite (banco de dados);
6. Módulo GSM/GPRS SIM900;
7. Módulo GPS NEO-6M;
8. Tiva TM4C123G LaunchPad;
9. IDE Energia.

4.2.1 IDE Energia

O Energia é uma plataforma de prototipagem eletrônica *open-source* iniciada em 2012, que surgiu com objetivo trazer as facilidades do Wiring para programar o MSP30 LaunchPad, desenvolvido pela Texas Instruments®.

O Wiring é um *framework* de programação que foi desenvolvido para que usuários avançados, intermediários e iniciantes ao redor do mundo pudessem compartilhar suas ideias, conhecimentos e experiências coletivamente (WIRING, 2014).

Atualmente, o Energia já fornece a possibilidade de trabalhar com as seguintes plataformas:

- MSP430F5529 LaunchPad;
- MSP430 LaunchPad;
- MSP430 FraunchPad;
- C2000 LaunchPad;
- Stellaris LaunchPad;
- Tiva C Series LaunchPad;

4.2.2 IDE Eclipse

Esta IDE começou a ser desenvolvida em 1998 pela IBM, que devido a dificuldade em achar parceiros comerciais optou em deixar a IDE como *open source*. (ECLIPSE, 2013).

O Eclipse é uma IDE conhecido mais comumente para desenvolvimento em Java, no entanto, por meio de plug-ins, pode ser usado para desenvolver aplicações em várias linguagens, como C/C++, Python, PHP e a plataforma Android (ECLIPSE, 2013).

Para o desenvolvimento Android na IDE Eclipse, o plugin Android Developers Tools (ADT) deve ser instalado. Este foi projetado para fornecer um ambiente poderoso e integrado para criação de aplicativos em Android. O ADT amplia os recursos do Eclipse para que o desenvolvedor possa rapidamente criar novos projetos, criar uma interface para os aplicativos, adicionar pacotes com base nas APIs do Android, depurar aplicativos utilizando as ferramentas do SDK do Android e até mesmo distribuir as aplicações (ANDROID, 2013c).

Os aplicativos Android no Eclipse foram codificados na linguagem Java. O Java é uma linguagem de programação orientada a objetos, tendo sido desenvolvida na década de 90 pela Sun Microsystems, posteriormente adquirida pela empresa Oracle.

O Java vem mantendo a posição de plataforma de desenvolvimento de aplicativos mais utilizada no mundo (SANTANA, 2013), fato que se deve à suas diversas vantagens, tais como:

- Independência de plataforma: é executado na maioria dos hardwares e nos principais sistemas operacionais;

- É um ambiente de desenvolvimento gratuito, e conta com várias IDEs para o desenvolvimento;
- Possui muitos materiais a respeito na Internet para auxiliar os desenvolvedores.

4.2.3 SQLite

O SQLite (2013) é um repositório de dados que simula um banco de dados relacional. Ela é a principal forma de persistência em aplicativos Android.

Este é formado por um conjunto de bibliotecas escritas em C, podendo ser integrado a programas escritos em diferentes linguagens com o intuito de possibilitar a manipulação de dados através de instruções SQL. A diferença para os outros bancos é que tudo isso pode ser feito sem que seja preciso acessar um SGBD (Sistema de Gerenciamento de Banco de Dados), ou seja, todas as instruções podem ser executadas diretamente via código fonte, como criação do banco e de tabelas, sendo por este motivo adotado como o banco de dados nativo da plataforma Android.

Na prática, o SQLite é um mini-SGBD, capaz de criar um arquivo em disco e ler e escrever diretamente. O SQLite não necessita de instalação, configuração ou administração, suporta *commit* e *rollback*, além de ser um banco de dados gratuito.

4.2.4 Android SDK

Para facilitar o desenvolvimento de aplicativos para a plataforma, a Google disponibiliza gratuitamente um ambiente de desenvolvimento para o Android, chamado de Android SDK (*Software Development Kit*), que fornece biblioteca e ferramentas necessárias para desenvolver e testar aplicativos para o Android.

O Android SDK permite que os desenvolvedores elaborem as aplicações a partir de um dispositivo virtual, este emula um *smartphone* ou um *tablet*, emulando grande parte das funções oferecidas pelos aparelhos, como *touchscreen*, câmera, GPS, acelerômetro, entre outras (ANDROID, 2013b).

Embora deva estar instalado na máquina em que se deseja programar para dispositivos Android, o Android SDK não possui uma IDE de desenvolvimento, sendo apenas um Kit de Desenvolvimento Padrão. Por esse motivo, ele deve ser integrado via plug-in com IDEs de desenvolvimento.

O *Android Virtual Device*, cujo acrônimo é AVD, é um emulador do sistema operacional Android que permite ao desenvolvedor simular um dispositivo real, definindo opções de hardware e software a serem seguidos pelo emulador Android.

A AVD consiste em um perfil de hardware no qual se define as características de hardware (por exemplo, se o dispositivo possui câmera, teclado QWERTY, etc.); um mapeamento da imagem do sistema, o qual é definido a versão da plataforma Android que será emulado no dispositivo virtual (por exemplo, Android 2.3 e Android 4.0); uma área de armazenamento dedicado em sua máquina ou emulador um cartão SD; além de opções como controlar dimensões da tela, aparência e assim por diante (ANDROID, 2012e). O AVD pode ser visualizado na Figura 19.



Figura 19 – Android Virtual Device 4.2.2
Fonte: Android (2012e).

4.2.5 Google Maps API

O Android possui extensões que possibilitam a criação de ambientes de desenvolvimento usando bibliotecas externas específicas, como por exemplo, o Google APIs (*Application Programming Interface*), ou componentes extras.

As APIs do Google possuem acesso aos serviços e dados do mesmo. Essas APIs fazem parte do SDK, ou seja, a partir do SDK é possível também obter um SDK de APIs do Google para Android. Um recurso importante dessas APIs é a biblioteca externa de mapas, que fornece uma API para que aplicativos Android possam ter acesso ao Google Maps,

permitindo a adição de recursos avançados de mapeamento, onde pode-se exibir um ou mais locais no mapa para um usuário.

A API do Google Maps para Android oferece aos desenvolvedores os meios para criar aplicativos com a funcionalidade de localização. A versão 2 da API do Google Maps foi lançado no final de 2012 e introduziu uma série de novos recursos, incluindo 3D, melhor cache, entre outros (ANDROID, 2013d).

Para poder acessar o Google Maps através de um aplicativo, é preciso obter uma chave de registro na API do Google Maps, a qual é gratuita e fácil de adquirir, porém, esta chave é única para cada desenvolvedor.

5 PROJETO DESENVOLVIDO

Esta seção mostra com mais detalhes como deverá funcionar o sistema, sendo dividida em duas subseções, uma abordando a parte de software e outra a parte de hardware.

5.1 Projeto de software

Nesta seção são apresentados quadros de requisitos, diagramas de casos de uso, diagramas de classe e Diagramas Entidade-Relacionamento, para fornecer uma visão melhor do sistema.

5.1.1 Levantamento de requisitos

O levantamento de requisitos é uma das partes mais importantes do processo que resultará no desenvolvimento de um sistema. Entender do que se trata o sistema, as regras que serão adotadas por ele são de extrema importância (SOMMERVILLE, 2007).

Nos quadros que seguem, são apresentados os requisitos funcionais e não funcionais identificados no presente projeto.

F1 Ativar alarme		Oculto ()		
Descrição: O usuário irá ativar o alarme através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF1.1 Ativar o alarme	O alarme do veículo somente será ativado onde houver a cobertura da operadora.	Confiabilidade	()	(X)

Quadro 3 – Requisito ativar alarme

F2 Ativar trava elétrica		Oculto ()		
Descrição: As travas elétricas das portas do veículo serão desativadas pelo usuário através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF2.1 Ativar trava automaticamente	As travas elétricas das portas do veículo serão ativadas automaticamente quando o alarme do mesmo for ativado.	Confiabilidade /Segurança	()	(X)

Quadro 4 – Requisito ativar trava elétrica

F3 Levantar vidro elétrico		Oculto ()		
Descrição: Os vidros elétricos das portas do veículo serão levantados pelo usuário através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF3.1 Ativar vidros automaticam ente	Os vidros elétricos das portas do veículo serão levantados automaticamente quando o alarme do mesmo for ativado.	Confiabilidade /Segurança	()	(X)

Quadro 5 – Requisito levantar vidro elétrico

F4 Desativar alarme		Oculto ()		
Descrição: O usuário irá desativar o alarme através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF4.1 Desativar o alarme	O alarme do veículo somente será desativado onde houver a rede da operadora.	Confiabilidade	()	(X)

Quadro 6 – Requisito desativar alarme

F5 Desativar trava elétrica		Oculto ()		
Descrição: As travas elétricas das portas do veículo serão desativadas pelo usuário através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF5.1 Desativar trava automaticam ente	As travas elétricas das portas do veículo serão desativadas automaticamente quando o alarme do mesmo for desativado.	Confiabilidade /Segurança	()	(X)

Quadro 7 – Requisito desativar trava

F6 Baixar vidro elétrico		Oculto ()		
Descrição: Os vidros elétricos das portas do veículo serão baixados pelo usuário através do aplicativo.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF6.1 Desativar trava automaticam ente	Os vidros elétricos das portas do veículo serão abaixados automaticamente quando o alarme do mesmo for desativado.	Confiabilidade /Segurança	()	(X)

Quadro 8 – Requisito baixar vidro elétrico

F7 Rastreamento		Oculto (X)		
Descrição: O sistema irá realizar o rastreamento ininterruptamente.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente
NF6.1 Rede da operadora	O sistema somente irá realizar o rastreamento da posição atual do veículo onde houver cobertura da rede da operadora.	Confiabilidade	()	(X)
NF6.2 Armazenamento	O sistema irá armazenar em um banco de dados do <i>smartphone</i> as últimas posições do veículo.	Confiabilidade	()	(X)
NF6.3 Envio das posições	O envio da posição do veículo é enviado a cada 10 minutos para o dispositivo móvel do usuário.			

Quadro 9 – Requisito rastreamento

F8 Executar ações no veículo		Oculto (X)		
Descrição: O microcontrolador enviará um sinal para ativar/desativar determinado recurso solicitado pelo usuário.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente

Quadro 10 – Requisito executar ações no veículo

F8 Verificar estado do alarme		Oculto (X)		
Descrição: O usuário poderá verificar se o alarme está ativado ou desativado.				
Requisitos Não-Funcionais				
Nome	Restrição	Categoria	Desejável	Permanente

Quadro 11 – Requisito verificar estado do alarme.

5.1.2 Diagramas de casos de uso

Diagramas de casos de uso descrevem relacionamentos e dependências entre os atores e as funcionalidades do sistema. Este tipo de diagrama não descreve os mecanismos internos de um sistema, e sim, tem a função de facilitar a comunicação com futuros usuários do sistema, e são especialmente úteis para determinar os recursos necessários que um sistema deve ter (SOMMERVILLE, 2007).

No Quadro 12 é apresentado uma breve descrição dos casos de uso do presente projeto.

Nome	Atores	Descrição	Referências Cruzadas
Ativar alarme	Usuário	O usuário irá ativar o alarme do veículo através de um comando do seu <i>smartphone</i> e, juntamente com o alarme, as travas elétricas e vidros elétricos também serão ativados.	F2, F3
Desativar alarme	Usuário	O usuário irá desativar o alarme do veículo através de um comando do seu <i>smartphone</i> e, juntamente com o alarme, as travas elétricas e vidros elétricos também serão desativados.	F5, F6
Levantar vidros elétricos	Usuário	O usuário terá o controle para fechar as janelas do seu veículo.	F4
Abaixar vidros elétricos	Usuário	O usuário terá o controle para abrir as janelas do seu veículo.	F4
Ativar travas elétricas	Usuário	O usuário terá controle para trancar as portas do seu veículo.	F4
Desativar travas elétricas	Usuário	O usuário terá controle para destrancar as portas do seu veículo.	F4
Verificar posição do veículo	Usuário	O usuário poderá verificar a atual posição do seu veículo.	F7
Verificar estado do alarme	Usuário	O usuário poderá verificar se o alarme do veículo está ativado ou desativado.	F8
Executar ações	Microcontrolador	O microcontrolador será responsável por executar as ações no veículo relativas aos recursos como ativação/desativação do alarme, trancar/destrancar as portas, abrir/fechar as janelas, as quais são requisitadas pelo através do seu <i>smartphone</i> .	F1, F2, F3, F4, F5, F6

Quadro 12 – Casos de uso

A Figura 20 apresenta o diagrama de caso de uso do sistema.

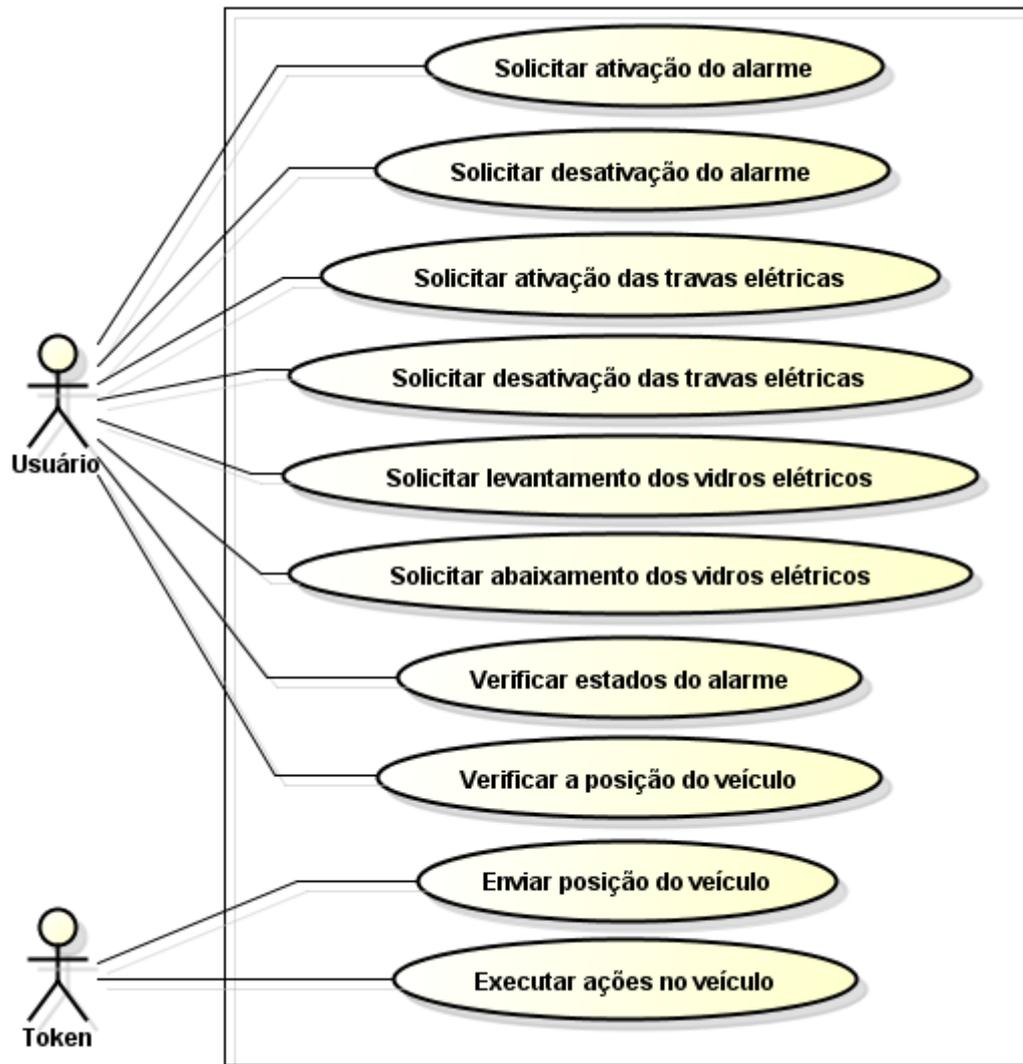


Figura 20 – Diagrama de caso de uso do sistema

Fonte: Autoria própria.

O Token apresentado na Figura 20 se trata dos componentes que ficam no interior do veículo, que são: microcontrolador, módulo GPS, módulo GSM, servo motor, motor DC e sensor PIR.

Na sequência são apresentadas as expansões dos casos de uso citados.

Caso de Uso: ativar alarme.
Atores: usuário.
Interessados: usuário.
Pré-condições: o usuário irá se sair/afastar do veículo.
Pós-condições: Após o alarme ter sido adicionado, tanto as travas elétricas quanto os vidros

elétricos também serão acionados.
Requisitos Correlacionados: F2, F3
<p>Fluxo principal:</p> <ol style="list-style-type: none"> 1. O usuário acessa o aplicativo no seu <i>smartphone</i>. 2. O usuário seleciona a opção para ativar o alarme. 3. O aplicativo apresenta uma mensagem dizendo que o alarme foi ativado.
<p>Tratamento de exceções:</p> <ol style="list-style-type: none"> 1. Caso o usuário se afaste do veículo e o mesmo não ative o alarme, este se ativará automaticamente caso o usuário se afaste a uma distância pré-determinada. 2. O alarme não se ativará caso o veículo esteja com as portas abertas. 3. O alarme não se ativará caso os vidros das portas do veículo estejam abertos.

Quadro 13 – Casos de uso – Ativar alarme

Caso de Uso: desativar alarme.
Atores: usuário.
Interessados: usuário.
Pré-condições: o usuário desejará entrar no veículo.
Pós-condições: O alarme será desativado.
Requisitos Correlacionados: F5, F6
<p>Fluxo principal:</p> <ol style="list-style-type: none"> 1. O usuário acessa o aplicativo no seu <i>smartphone</i>. 2. O usuário seleciona a opção para desativar o alarme. 3. O aplicativo apresenta uma mensagem dizendo que o alarme foi desativado.
<p>Tratamento de exceções:</p> <ol style="list-style-type: none"> 1. O usuário somente terá o controle dos vidros elétricos e das travas elétricas das portas do veículo se o alarme do mesmo estiver desativado.

Quadro 14 – Casos de uso – Desativar alarme

Caso de Uso: levantar vidros elétricos
Atores: usuário.
Interessados: usuário.
Pré-condições: o alarme do carro deverá estar desativado.
Pós-condições: O vidro será levantado quando o usuário optar por fechar a janela da porta do veículo.

Requisitos Correlacionados: F4
Fluxo principal: 1. O usuário acessa o aplicativo no seu <i>smartphone</i> . 2. O usuário seleciona a opção para fechar as janelas do veículo. 3. O aplicativo apresenta uma mensagem dizendo que os vidros foram levantados.
Tratamento de exceções:

Quadro 15 – Casos de uso – Levantar vidros elétricos

Caso de Uso: descer vidros elétricos
Atores: usuário.
Interessados: usuário.
Pré-condições: o alarme do carro deverá estar desativado.
Pós-condições: O vidro será abaixado quando o usuário optar por abrir a janela da porta do veículo.
Requisitos Correlacionados: F4
Fluxo principal: 1. O usuário acessa o aplicativo no seu <i>smartphone</i> . 2. O usuário seleciona a opção para abrir as janelas do veículo. 3. O aplicativo apresenta uma mensagem dizendo que os vidros foram abaixados.
Tratamento de exceções:

Quadro 16 – Casos de uso – Descer vidros elétricos

Caso de Uso: ativar travas elétricas.
Atores: usuário.
Interessados: usuário.
Pré-condições: o alarme do carro deverá estar desativado.
Pós-condições: A trava elétrica das portas será ativada quando o usuário optar por trancar as portas do veículo.
Requisitos Correlacionados: F4
Fluxo principal: 1. O usuário acessa o aplicativo no seu <i>smartphone</i> . 2. O usuário seleciona a opção para trancar as portas do veículo. 3. O aplicativo apresenta uma mensagem dizendo que as portas foram trancadas.

Tratamento de exceções:**Quadro 17 – Casos de uso – Ativar travas elétricas**

Caso de Uso: desativar travas elétricas.

Atores: usuário.

Interessados: usuário.

Pré-condições: o alarme do carro deverá estar desativado.

Pós-condições: A trava elétrica das portas será desativada quando o usuário optar por destrancar as portas do veículo.

Requisitos Correlacionados: F4

Fluxo principal:

1. O usuário acessa o aplicativo no seu *smartphone*.
2. O usuário seleciona a opção para destrancar as portas do veículo.
3. O aplicativo apresenta uma mensagem dizendo que as portas foram destrancadas.

Tratamento de exceções:**Quadro 18 – Casos de uso – desativar travas elétricas**

Caso de Uso: Verificar a posição do veículo.

Atores: usuário.

Interessados: usuário.

Pré-condições: O *smartphone* cliente deve estar se comunicando com o módulo GSM.

Pós-condições: Será apresentada para o usuário uma lista das últimas posições do seu veículo.

Requisitos Correlacionados:**Fluxo principal:**

1. O usuário acessa o aplicativo no seu *smartphone*.
2. O usuário seleciona a opção para verificar uma lista das posições do seu veículo.
3. O aplicativo apresenta uma lista referentes as últimas posições do veículo.
4. O usuário poderá visualizar em um mapa cada posição do seu veículo.

Tratamento de exceções:

1. A captura da posição do veículo somente será possível quando o mesmo se encontrar onde houver cobertura da rede da operadora.

Quadro 19 – Casos de uso – Verificar posição do veículo

Caso de Uso: executar as ações no veículo.
Atores: Microcontrolador.
Interessados: Microcontrolador.
Pré-condições: O microcontrolador deve estar conectado com o módulo GPS e o módulo GSM.
Pós-condições: Será executado determinada ação sobre algum recurso solicitado pelo usuário.
Requisitos Correlacionados: F1, F2, F3, F4, F5, F6
Fluxo principal: 1. O microcontrolador enviará um sinal para ativar determinado recurso solicitado pelo usuário.
Tratamento de exceções:

Quadro 20 – Casos de uso – Executar as ações no veículo

Caso de Uso: Verificar estado do alarme
Atores: Usuário.
Interessados: Usuário.
Pré-condições: O dispositivo móvel do usuário deverá estar conectado ao <i>Token</i> dentro do veículo.
Pós-condições: Será exibido na interface do aplicativo Android o estado do alarme do veículo.
Requisitos Correlacionados: F1, F2, F3, F4, F5, F6
Fluxo principal: 1. Ao ativar o alarme do veículo será exibido no aplicativo Android do dispositivo móvel que o mesmo está ativo. 2. Ao desativar o alarme do veículo será exibido no aplicativo Android do dispositivo móvel que o mesmo está desativado.
Tratamento de exceções:

Quadro 21 – Casos de uso – Verificar estado do alarme

5.1.3 Diagrama de Classes

O diagrama de classes é consequência do prévio levantamento de requisitos, definição dos casos de uso e classes. Ele lista todos os conceitos de domínio que serão implementados no sistema e suas relações, se tornando importante, pois define a estrutura do sistema a ser desenvolvido (SOMMERVILLE, 2007).

A Figura 21 representa o diagrama de classes do presente projeto.

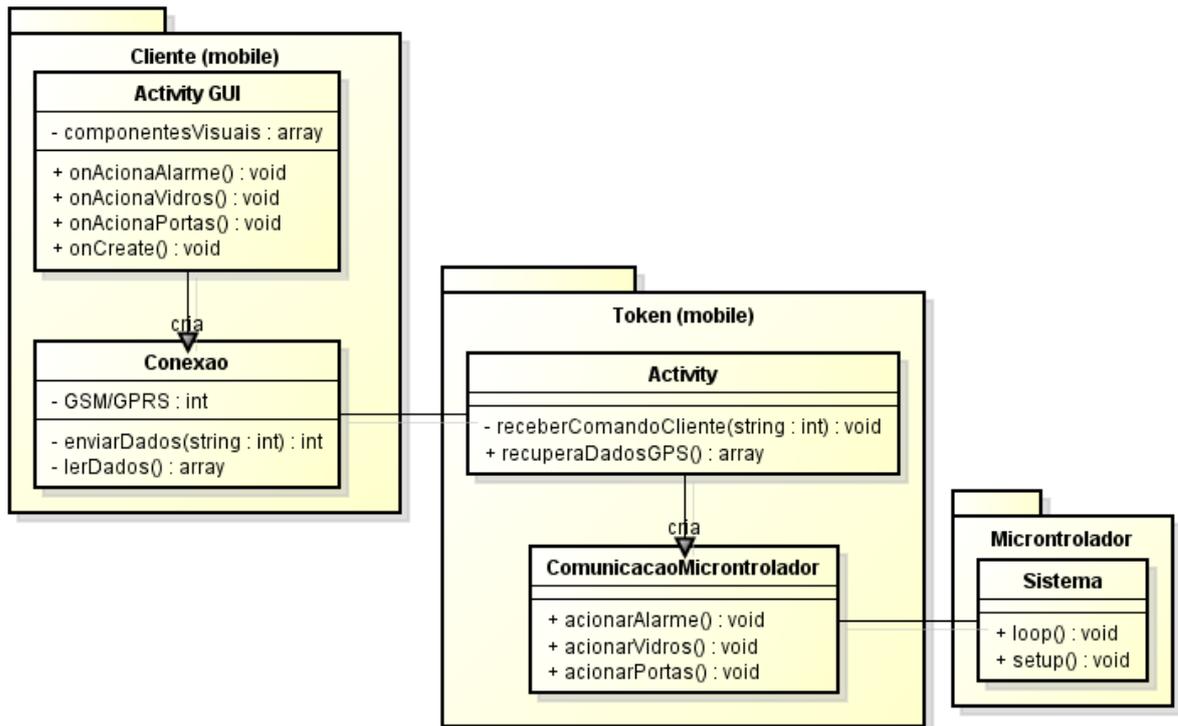


Figura 21 – Diagrama de Classes

Fonte: Autoria própria.

5.1.4 Diagrama Entidade-Relacionamento

O diagrama entidade relacionamento é um modelo diagramático que descreve o modelo de dados do sistema com alto nível de abstração (SOMMERVILLE, 2007).

A Figura 22 representa o diagrama entidade relacionamento do presente projeto.

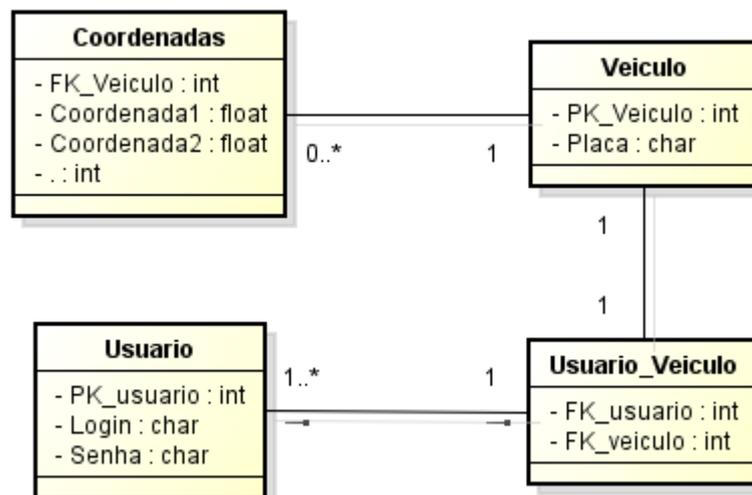


Figura 22 – Diagrama Entidade-Relacionamento

Fonte: Autoria própria.

As coordenadas referentes ao posicionamento do veículo serão armazenados no banco de dados configurado no próprio aplicativo, onde o usuário terá acesso a essas informações.

5.2 Projeto de hardware

Nesta seção é descrito como foi feita a integração entre hardware e software.

5.2.1 Aplicativo

Diante das mais variadas soluções de mercado existentes na área, este software é concebido sobre uma estratégia inovadora que permite implantação rápida para o rastreamento de um veículo e controle sobre algumas funcionalidades do mesmo, que são controle do alarme, vidros elétricos e travas elétricas.

A proposta deste software segue a construção elaborada em três vertentes:

1. A implantação de um hardware que possibilite enviar dados remotamente de um veículo para um *smartphone* e vice-versa.
2. A concepção de um software que permita ao usuário realizar o controle das funcionalidades já especificadas.
3. A integração dos sistemas citados acima, para que operando conjuntamente possam idealizar e produzir uma plataforma útil e funcional.

5.2.2 Arquitetura do sistema

A estrutura e funcionamento se dá de acordo com a Figura 23, esta composta de um módulo GSM, um módulo GPS e um microcontrolador, localizado no interior do veículo, que fornece dados de posicionamento e recebe comandos para a realização de alguma ação relacionada ao alarme, vidros elétricos e travas elétricas do *smartphone* cliente que fica em posse do usuário. Com posse dos dados referente ao posicionamento do veículo, os mesmos são enviados através de um SMS para o *smartphone* do usuário.

O módulo GPS fica constantemente coletando dados de sua posição global e a cada intervalo de tempo os mesmos são enviados para o *smartphone* do usuário para que o mesmo possa verificar o posicionamento do veículo em tempo real. Caso o veículo e o receptor GPS se encontre em um local em que seja impossível a recepção das coordenadas geográficas,

como, por exemplo, dentro de túneis ou garagens cobertas com zinco e/ou o *smartphone* esteja em um local onde não haja a cobertura da rede da operadora, o sistema ficará inoperante.

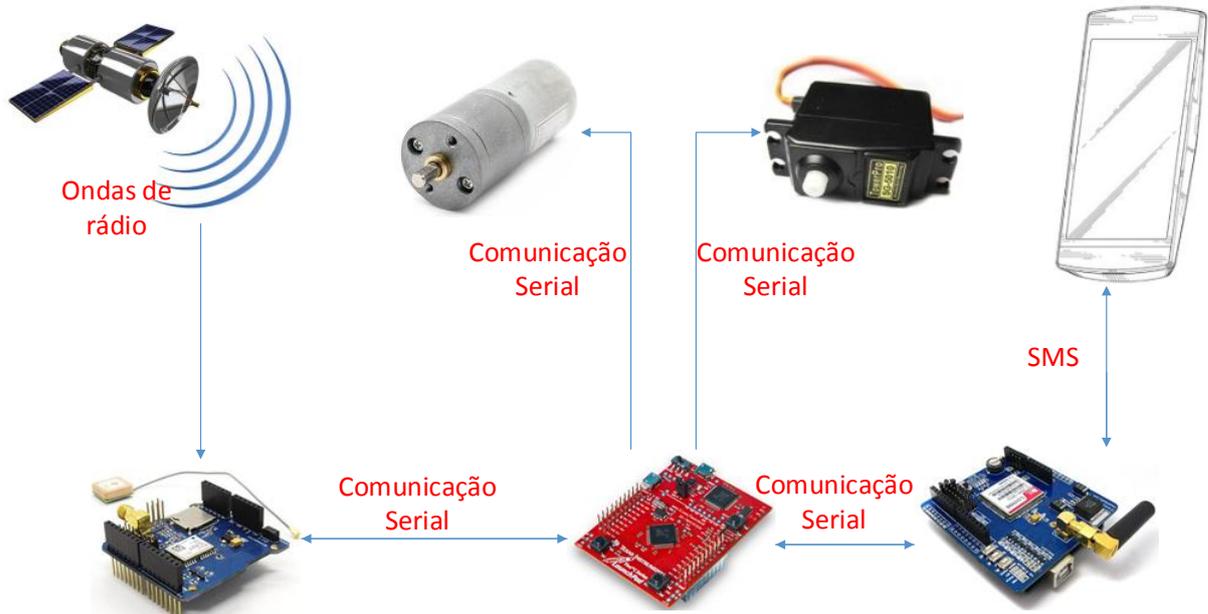


Figura 23 – Arquitetura do sistema

Fonte: Autoria própria.

A Figura 23 representa conceitualmente o processo de aquisição de informações, transmissão de dados, processamento, aplicação e acesso ao usuário. Abaixo, descreve-se a mesma:

- Aquisição de informações: As informações apuradas pelo receptor GPS são basicamente os dados de posicionamento do mesmo, que será transmitido via SMS.
- Transmissão de dados: Ocorre através da troca de SMS entre o módulo GSM e o *smartphone*.
- Processamento: Quando se trata do rastreamento, a aplicação do *smartphone* do usuário receberá uma SMS, processará a mesma e com os dados referentes ao posicionamento mostrará em uma mapa a posição do veículo. Quando se trata da ativação/desativação dos recursos do veículo o usuário enviará uma SMS para o módulo GSM a qual se comunicará com o microcontrolador que será responsável pela comunicação com outros periféricos via serial executando as ações desejadas.
- Aplicação: Podendo ser acessado remotamente, a aplicação exibe ao usuário uma interface visual que fornece um mapa cartográfico que possibilita a visualização de uma determinada posição escolhida pelo usuário. Além de fornecer uma série de

outros recursos, como o controle referente a ativação/desativação do alarme, travas elétricas e vidros elétricos do veículo.

5.2.3 Funcionalidade do sistema

O software fornece uma gama de recursos e ferramentas que processam os dados, transformando estes em informações úteis para o usuário.

As funcionalidades que o aplicativo oferece são:

1. **Localização:** Através do *smartphone* cliente o usuário poderá solicitar a localização geográfica imediata do veículo.
2. **Últimas posições:** É armazenado em um banco de dados um histórico das posições do veículo, podendo exibir estas em um mapa.
3. **Controle do alarme do veículo:** o usuário poderá ativar/desativar o alarme do veículo através do aplicativo e, juntamente com o alarme, as travas elétricas e vidros elétricos também serão acionados automaticamente de acordo com o comando efetuado. Além disso, tanto as travas elétricas quanto os vidros elétricos poderão ser controlados individualmente.

6 RESULTADOS E TESTES DE VALIDAÇÃO

Neste capítulo apresenta-se a proposta do protótipo para rastreamento e controle de alguns recursos do veículo através de um *smartphone*. Também é abordado sobre a forma de obtenção de dados para o funcionamento do sistema proposto.

6.1 Desenvolvimento do protótipo

O microcontrolador foi ligado ao módulo GSM, ao módulo GPS e aos motores para que pudesse a comunicação entre os mesmos. Este esquema de montagem é apresentado na Figura 24.

O sensor de presença possui três terminais: um para alimentação, um terra e outro para envio de sinal caso haja movimento em sua área de alcance. Os pinos de alimentação e terra foram ligados aos pinos correspondentes do microcontrolador, já o pino de sinal foi ligado a porta de entrada PE_1 do microcontrolador.

Assim como o sensor de presença, o servo motor também conta com três terminais, sendo um deles para alimentação, outro terra e outro para receber a informação referente ao seu giro (ângulo). Os pinos de alimentação e terra também foram conectados aos pinos correspondentes do microcontrolador e o pino de sinal foi ligado a porta PE_3.

Para o acionamento do motor DC, verificou-se que a tensão fornecida pelo microcontrolador não era suficiente, já que o mesmo fornece tensão de 3,3V e o motor DC precisa de 12V, assim, foi usada uma fonte externa que fornece a voltagem necessário para o acionamento do mesmo. Para que fosse possível girar o motor em ambos os sentidos (horário e anti-horário) foi usado o circuito integrado L293 no qual a entrada de sinais 1 (*input 1*) foi ligada a porta PA_3 e a entrada de sinais 2 (*input 2*) foi ligada a porta PA_2 e as saídas do L293 foram ligados no motor DC.

Para receber as coordenadas geográficas do módulo GPS, o pino de transmissão de dados (TX) do módulo foi ligado ao pino PE_4, que é o pino de recepção de dados (RX) da porta serial 5 do microcontrolador e o pino de RX de dados do módulo GPS foi ligado ao pino PE_5, que é o pino de TX de dados da porta serial 5 do microcontrolador, ou seja, é através da porta serial 5 do microcontrolador que o mesmo se comunica com o módulo GPS.

Finalizando, para efetuar a comunicação entre o módulo GSM e o microcontrolador foi utilizado a porta serial 3 deste último, ou seja, o pino de TX do módulo GSM foi ligado a porta PC_6 do microcontrolador, que é o pino de RX da porta serial 3 do mesmo e o pino de

RX do módulo foi ligado a porta PC_7 do microcontrolador, que é o pino de TX da porta serial 3 do microcontrolador.

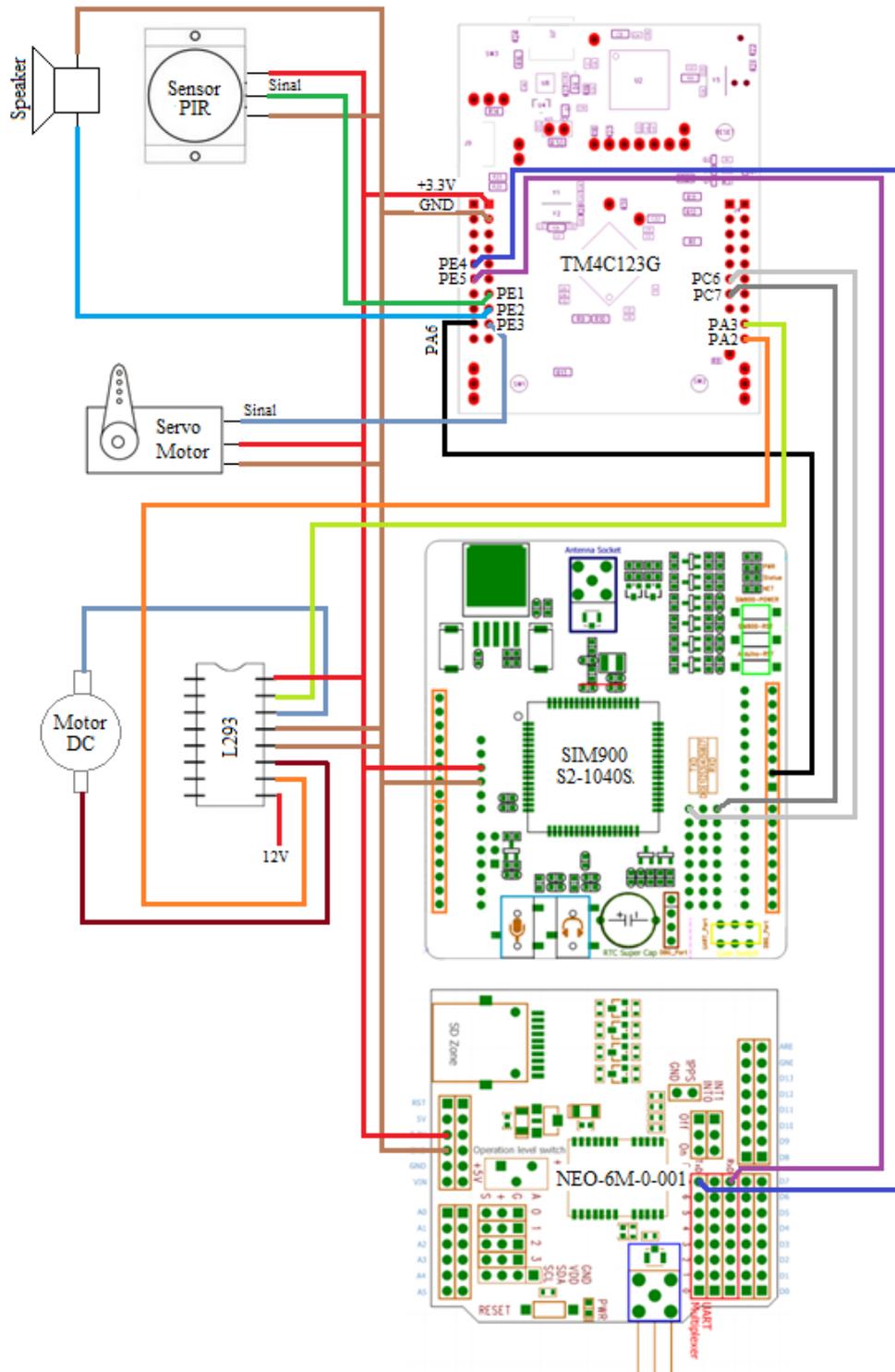


Figura 24 – Esquema de montagem dos componentes do protótipo
 Fonte: Autoria própria.

6.2 Programação do Microcontrolador

O passo inicial do programa era estabelecer a conexão entre o microcontrolador e os periféricos (módulo GPS, módulo GSM, motores e sensor PIR). Para isso o módulo GPS foi ligado a porta serial 5 do microcontrolador, que é por onde este último recebe e trata as coordenadas geográficas. O módulo GSM foi ligado a porta serial 3 do microcontrolador, pela qual é recebida as informações que o mesmo deve tratar para executar determinada ação e também usada para transmitir os comandos AT, responsáveis pelo envio de um SMS contendo as informações referente a posição do receptor GPS.

Todos os comandos AT são transmitidos através da porta serial 3. Para configurar o módulo GSM para trabalhar com SMS é enviado o comando:

```
Serial3.println("AT+CMGF=\r");
```

Para preparar o módulo para receber SMS não lida é utilizado o comando:

```
Serial3.println("AT+CMGL=\"REC UNREAD\"\r");
```

De acordo com a mensagem recebida será executada a ação de acordo com o Quadro 22.

SMS	Ação
Ativa_Alarme	Ativa o alarme (sensor PIR)
Des_Alarme	Desativa o alarme (Sensor PIR)
Ativa_Trava	Ativa trava elétrica (servo motor)
Des_Trava	Desativa trava elétrica (servo motor)
Fechar_Vidro	Fecha vidro elétrico (motor DC)
Abrir_Vidro	Abre vidro elétrico (motor DC)

Quadro 22 – Ação a ser executada de acordo com a SMS recebida pelo módulo GSM

O comando AT utilizado para enviar um SMS para determinado número é:

```
Serial3.println("AT+CMGS=\"+554691064999\"\r");
```

A mensagem SMS enviada é para o dispositivo móvel contém a informação relativa as coordenadas recebidas do módulo GPS após a leitura de sua respectiva porta serial, que são no formato, por exemplo: 2613.47199S05240.07850W. Esta mensagem é tratada no aplicativo Android para separar corretamente a latitude e a longitude.

Após recebida a SMS e executada a ação por ela definida a mesma é excluída com o comando:

```
Serial3.println("AT+CMGD=1,2\r");
```

O comando CMGD espera por parâmetro um índice que especifica em o local da memória e uma *flag* que é a ação a ser tomada naquele local. Neste caso o “1” seleciona toda a memória onde estão armazenadas as SMS recebidas e a *flag* “2” deleta todas as SMS lidas.

6.3 Aplicativo Android

O aplicativo Android consiste em apresentar as últimas posições do veículo de propriedade do usuário e também enviar comandos via SMS para o módulo GSM presente no veículo que seriam para ativar/desativar o alarme, ativar/desativar as travas elétricas e subir/descer os vidros elétricos.

O projeto é dividido em quatro classes, a classe principal, a classe responsável por apresentar uma lista das posições do veículo para o usuário, a classe para tratar os mapas que tem como função apresentar um mapa no *display* do *smartphone* e uma classe responsável pelo recebimento de SMS.

6.3.1 Classe Principal

A interface inicial da aplicação, mostrada para o usuário é representada pela Figura 25.

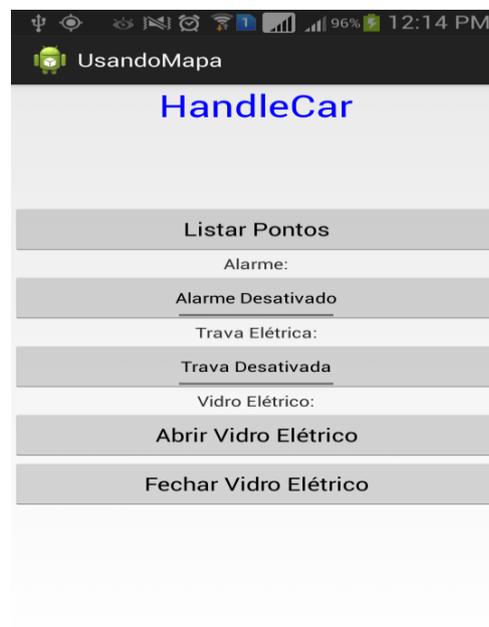


Figura 25 – Tela Inicial do aplicativo Android
Fonte: Autoria própria.

A primeira ação a ser feita na classe principal é a criação do banco de dados (Listagem 1) que será responsável por armazenar as coordenadas geográficas recebidas do receptor GPS localizado no interior do veículo.

Este banco de dados é criado na primeira execução do aplicativo, nas demais, ele é apenas recuperado, conforme apresentado.

```
1:   bd = this.openOrCreateDatabase("dadosRastreamento", Context.MODE_PRIVATE,
null);
2:   bd.execSQL("CREATE TABLE IF NOT EXISTS localizacao ( _id INTEGER PRIMARY
KEY, " + "latitude DECIMAL, longitude DECIMAL, datahora DATE )");
```

Listagem 1 - Criação do banco de dados SQLite

A linha 1 é responsável por criar o banco de dados com o nome “dadosRastreamento” no modo privado, ou seja, só a aplicação terá acesso a ele, já a linha 2 cria uma tabela neste mesmo banco com o nome “localizacao” com os seguintes atributos: id (chave primária), latitude, longitude e a data e hora atual.

Estando o aplicativo em execução, o passo seguinte é deixar um processamento em paralelo para receber as mensagens SMS, verificar se a mesma foi enviada do módulo GSM do com o conteúdo referente a localização do veículo (latitude e longitude) e persistir tais informações, conforme apresentado na Listagem 2:

```
1:  (...)
2:  IntentFilter intentFilter = new IntentFilter("SmsMessage.intent.MAIN");
3:  mIntentReceiver = new BroadcastReceiver()
4:  (...)
5:  String msg = intent.getStringExtra("get_msg");
6:  (...)
7:  this.registerReceiver(mIntentReceiver, intentFilter);
8:  (...)
```

Listagem 2 - Recuperando conteúdo do SMS

A linha 2 cria um objeto do tipo IntentFilter que será responsável por capturar os dados do tipo SmsMessage.intent.MAIN que serão passados pela ClasseSMS, sendo que está é chamada a cada SMS recebida, logo na sequência, na linha 5 é recuperado o conteúdo do SMS e em seguida essa *string* é formatada para persistir os dados, como demonstrado na Listagem 3.

```
1:  Date data = new Date();
2:  SimpleDateFormat df = new SimpleDateFormat("dd/mm/yyyy hh:mm:ss");
3:  String dataf = df.format(data);
4:  double latitude = Double.parseDouble(tvLatitude.getText().toString());
5:  double longitude = Double.parseDouble(tvLongitude.getText().toString());
  (...)
6:  ContentValues registro = new ContentValues(); //Seta valores na tabela
7:  registro.put("latitude", latitude);
```

```

8: registro.put("longitude", longitude);
9: registro.put("datahora", dataf);
10: bd.insert("localizacao", null, registro); //Inserere na tabela

```

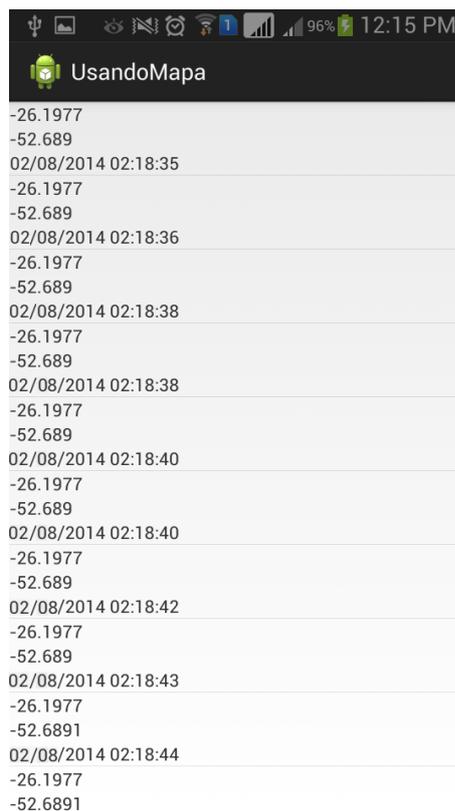
Listagem 3 - Inserção das informações no banco de dados

As três primeiras linhas são responsáveis por adquirir a data atual do sistema para ser apresentado para o usuário em que momento determinada posição foi recebida.

As linhas 4 e 5 convertem os dados, que antes eram *string*, de latitude e longitude para *double*, para que nas linhas 6 à 10 estes dados sejam adicionados no banco de dados.

6.3.2 Classe responsável pela lista de posições

Está lista é apresentada quando o usuário optar por ver as últimas posições relativas a seu veículo conforme Figura 26. Vale ressaltar que a hora referente a cada posição do veículo apresentada para o usuário é recuperada do sistema do próprio dispositivo móvel.



-26.1977	-52.689	02/08/2014 02:18:35
-26.1977	-52.689	02/08/2014 02:18:36
-26.1977	-52.689	02/08/2014 02:18:38
-26.1977	-52.689	02/08/2014 02:18:38
-26.1977	-52.689	02/08/2014 02:18:40
-26.1977	-52.689	02/08/2014 02:18:40
-26.1977	-52.689	02/08/2014 02:18:42
-26.1977	-52.689	02/08/2014 02:18:43
-26.1977	-52.6891	02/08/2014 02:18:44
-26.1977	-52.6891	

Figura 26 – Lista de posições

Fonte: Autoria própria.

A Listagem 4 apresenta o código referente a lista.

```

1: public class Lista extends ListActivity {
2: private SQLiteDatabase bd;
3: private Cursor cursor;
4: private SimpleCursorAdapter adapter;
5: protected void onCreate(Bundle savedInstanceState) {
6:     super.onCreate(savedInstanceState);
7:     bd = openOrCreateDatabase("dadosRastreamento", Context.MODE_PRIVATE, null);
8:     cursor = bd.query("localizacao", new String[] {"_id", "latitude",
"longitude", "datahora"}, null, null, null, null, null);
9:     String[] colunas = new String[] {"latitude", "longitude", "datahora"};
10:    int campoTela[] = new int[] {R.id.tvLatitudeLista, R.id.tvLongitudeLista,
R.id.tvdatahoraLista};
11:    adapter = new SimpleCursorAdapter(this, R.layout.activity_lista, cursor,
colunas, campoTela);
12:    this.setListAdapter(adapter);
    }
13: public void onItemClick(ListView parent, View V, int position, long _id) {
14:    Cursor registro = (Cursor) this.getListAdapter().getItem(position);
15:    double lat = registro.getDouble( registro.getColumnIndex("latitude"));
16:    double longi = registro.getDouble( registro.getColumnIndex("longitude"));
17:    String dataHora = registro.getString( registro.getColumnIndex("datahora"));
18:    Intent i = new Intent( this, Mapa.class );
19:    i.putExtra( "longi", longi );
20:    i.putExtra( "lat", lat );
21:    i.putExtra( "datahora", dataHora );
22:    startActivity( i );
23: }

```

Listagem 4 - Activity responsável por apresentar a lista de posições para o usuário

Na linha 1 é estendida as funcionalidades de lista do Android. Logo na sequência, na linha 2 é declarado o banco de dados, na linha 3 um cursor que permite a de um conjunto de resultados retornados por uma consulta no banco de dados e na linha 4 é declarado um SimpleCursorAdapter, o qual é possível especificar qual coluna de uma tabela do banco de dados é desejado que seja exibido o resultado.

Na linha 7 o banco de dados criado na classe principal é aberto para que seja possível acessar a tabela “localização” que contém as informações de latitude, longitude e data e hora.

Da linha 13 a 22 é recuperado as informações relativas as posições, como latitude, longitude e data e hora. Estas informações também serão passadas para a classe responsável por apresentar a posição no mapa (linha 18).

6.3.3 Trabalhando com SMS em Android

O aplicativo Android permitirá a recepção de SMS, bem como seu processamento. Para se trabalhar com SMS em Android é preciso adicionar algumas permissões, estas apresentadas na Listagem 5.

```

1: <uses-permission android:name="android.permission.RECEIVE_SMS"/>
2: <uses-permission android:name="android.permission.READ_SMS" />

```

Listagem 5 - Permissões para trabalhar com SMS em Android

A primeira linha é a permissão para a aplicação monitorar o recebimento de SMS de outros dispositivos, já a linha 2 é para ler.

Há uma classe responsável por tratar o recebimento de SMS que é iniciada junto com a classe principal logo que o aplicativo é executado, que serão as coordenadas geográficas apresentadas em uma lista para o usuário acessá-las em um mapa. Esta é apresentada na Listagem 6.

```

1: public class ClasseSMS extends BroadcastReceiver {
2: public String corpoSMS;
3: public String numSMS;
4: public void onReceive( Context context, Intent intent ) {
5: Bundle extras = intent.getExtras();
6: if ( extras != null ) {
7: Object[] smsExtra = (Object[]) extras.get( "pdus" );
8: for ( int i = 0; i < smsExtra.length; i++ ) {
9: SmsMessage sms = SmsMessage.createFromPdu((byte[])smsExtra[i]);
10: corpoSMS = sms.getMessageBody().toString();
11: numSMS = sms.getOriginatingAddress();
12: Intent in = new Intent("SmsMessage.intent.MAIN").
        putExtra("get_msg",corpoSMS);
13: context.sendBroadcast(in);
14: this.abortBroadcast();
15: }
16: }
17: }
18:}

```

Listagem 6 - Classe que trata o recebimento de SMS

A primeira linha é estendida as funcionalidades da classe com “BroadcastReceiver”, que é um componente do Android que responde a determinados anúncios enviados ao sistema. O Android envia mensagens para todo o sistema quando determinados eventos acontecem, e esta mensagem pode ser respondida por quem tiver interesse em recebe-las. Um *broadcast* pode ser enviado tanto por aplicativos nativos do Android quando por aplicativos construído por desenvolvedores e através do *Broadcast Receiver* pode-se disparar determinadas ações quando um evento acontece.

Nas linhas 2 e 3 são declaradas duas *Strings* que será valorizada com o texto da SMS e com o número de quem enviou a mesma.

Dentro do método *onReceive*, na linha 5 é recuperado os dados recebidos e se houver dados, ou seja, diferente de vazio (linha 6) é verificado o conteúdo do SMS e separado o texto do SMS e o número do mesmo nas linhas 10 e 11.

Na sequência, as linhas 12 e 13 são responsáveis para passar a informação da SMS para outra classe, no caso, a classe principal e finalmente a linha 14 encerra o *broadcast*.

Para o envio de SMS do *smartphone* para o módulo GSM são utilizados os comandos apresentados na Listagem 7.

```
1: SmsManager smsManager = SmsManager.getDefault();
2: smsManager.sendTextMessage("+554699120676", null, "Trava Ativada", null, null);
```

Listagem 7 - Comandos para o envio de SMS

O `SmsManager` na linha 1 gerencia as operações que envolvem SMS e a linha 2 envia um SMS informando o número do destinatário e o seu conteúdo.

6.3.4 Trabalhando com mapas em Android

O primeiro passo na construção do aplicativo em Android foi preparar o ambiente para estender algumas funcionalidades da API do Google Maps, esta pode ser adquirida como parte do Google Play Services, sendo que este componente fornece funcionalidades essenciais, como autenticação para seus serviços do Google, serviços baseados na localização utilizando GPS e mapas, entre outros recursos.

Para permitir que o desenvolvedor utilize mapas em uma aplicação é preciso gerar um certificado digital que é utilizado para assinar o aplicativo Android. O Android SDK já traz uma *keystore* (chave) com sua instalação padrão. Esta chave é utilizada como *debug* e é acionada sempre quando um *smartphone* real foi conectado no PC/Notebook e testamos o aplicativo diretamente neste. Apesar do desenvolvedor não visualizar, o projeto foi compactado em um *apk* (*Android Package*) e assinado digitalmente, antes de ser carregado no dispositivo.

Com a chave em mãos, é possível gerar o *sha-1 fingerprint*, necessário para gerar uma chave para ser possível utilizar os recursos de mapas da API do Google Maps. Essa chave deve ser adicionada dentro do *Android Manifest*, de acordo com a Listagem 8.

```
<meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="API_KEY" />
```

Listagem 8 - Adicionando a chave no projeto para utilizar a Google Maps API

O arquivo “*androidmanifest.xml*” fica na raiz do projeto e é o arquivo principal do mesmo, contém todas as configurações necessárias para executar uma aplicação, como o nome do pacote utilizado, o nome das classes, as permissões de acesso a dados, entre outras.

Além da chave deve-se adicionar as permissões no arquivo “androidmanifest.xml”. A Listagem 9 demonstra a adição destas permissões.

```

1: <uses-permission android:name="android.permission.INTERNET"/>
2: <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
3: <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
4: <uses-permission
   android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
5: <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>

```

Listagem 9 - Permissões de usuário anexadas ao manifesto

A primeira linha é a permissão para o uso da internet, já a linha 2 permite que a aplicação acesse o estado de conectividade de rede do aparelho. Na linha 3 permite que o aplicativo leia dados em que estejam armazenados em outro servidor. A linha 4 permite ler dados dos serviços do Google e por fim, a linha 5 permite adquirir fontes para localização, como GPS, torres de celular e Wi-Fi.

O leiaute de interface gráfica usado para apresentar o mapa na tela no *smartphone* do usuário é mostrado na Listagem 10.

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <fragment xmlns:android="http://schemas.android.com/apk/res/android"
   android:id="@+id/mapas"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:name="com.google.android.gms.maps.MapFragment" />

```

Listagem 10 - Arquivo de definição do layout para apresentar o mapa na aplicação

O único componente na interface gráfica é um *fragment*. Neste caso, fez-se necessário a definição da propriedade *android:name*, para indicar qual a classe responsável pelo conteúdo do fragmento e fornecer conteúdo visual do mapa na tela.

A Listagem 11 mostra código da *activity* Mapa para recuperar uma posição e apresenta-la no mapa.

```

1: Intent i = getIntent();
2: double longi = i.getDoubleExtra( "longi", 0 );
3: double lat = i.getDoubleExtra( "lat", 0 );
4: String dataHora = i.getStringExtra( "datahora" );
5: GoogleMap map =
((MapFragment)getFragmentManager().findFragmentById(R.id.mapas)).getMap();
6: LatLng ponto = new LatLng(lat, longi);
7: map.moveCamera(CameraUpdateFactory.newLatLngZoom(ponto, 16));
8: map.addMarker(new MarkerOptions().title("Posicao").position(ponto).snippet(
dataHora));

```

Listagem 11 - Activity Mapa

As três primeiras linhas são responsáveis por recuperar as informações sobre a posição e data e hora adquiridas. A linha 5 recupera a instância de *MapFragment* inicialmente. Esta classe possui um método *getMap* que retorna a *GoogleMap*. Na linha 6 é criada uma instância de *LatLng* passando como parâmetro os valores de latitude e longitude, ambos são valores primitivos *double*. A linha 7 tem como função adicionar um zoom no valor estipulado no ponto de interesse e por último a linha 8 adiciona um marcador no mapa e este conta com um título, o ponto desejado e um *snippet* com a data e hora.

A Figura 27 mostra a tela de mapa apresentada para o usuário utilizando a Google Maps API.



Figura 27 – Mapa referente uma posição do veículo
Fonte: Autoria própria.

6.4 Testes e Verificação do Projeto

Após feita a integração de todas as partes do sistema (hardware e software) este foi testado para ver se iria se comportar como o esperado. A Figura 28 mostra como ficou o protótipo.

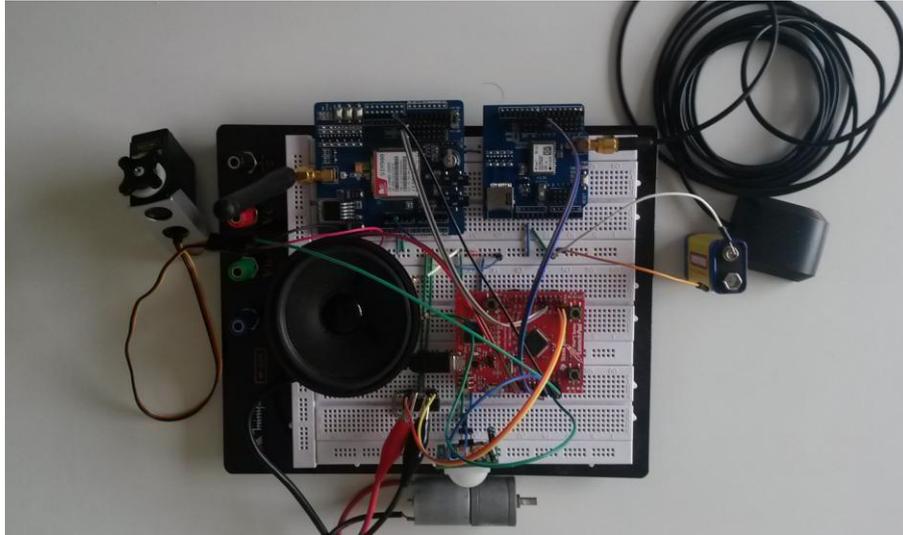


Figura 28 – Protótipo do projeto
 Fonte: Autoria própria.

Logo após ligar os componentes do sistema foi preciso aguardar alguns minutos até que o módulo GPS começasse a capturar as coordenadas do local. Após este tempo de espera verificou-se que uma SMS era enviada para o *smartphone* a cada dez minutos (tempo pré-definido no microcontrolador) com as coordenadas do receptor GPS e essas posições (armazenadas em um banco de dados) era mostrado em uma lista, como na Figura 26, a qual o usuário poderia selecionar uma determinada posição que esta seria mostrada em uma mapa, como apresentado na Figura 27.

Foi verificado que em caso de falhas com a cobertura da rede da operadora ou caso o receptor GPS esteja em um local inadequado para que o mesmo receba sua posição, o sistema fica fora de operação. Ao ser reestabelecida a conexão com a rede da operadora e que o receptor GPS se encontra em céu aberto, o sistema é capaz de voltar a funcionar normalmente, sem necessidade de reinicializar.

Ressalta-se que como o sistema é um protótipo, não há a possibilidade de fazer comparações com demais sistemas semelhantes apresentados na seção 2.6. Sendo assim o único objetivo do projeto é que este atendesse ao que lhe foi proposto:

- Possibilitar o usuário ter um aplicativo Android para controlar alguns recursos do veículos, que são: controle do alarme, travas elétricas e vidros elétricos.
- Coletar informações referentes ao posicionamento do veículo e enviá-las para o *smartphone* do usuário.

Com os testes foi possível verificar que os requisitos acima foram atendidos.

7 CONCLUSÃO

O grande número de dispositivos móveis, como *smartphones* e *tablets* existentes na atualidade gera, por pressões de seus usuários, uma crescente demanda por novas aplicações e serviços para seus dispositivos, fazendo com que os mesmos evoluam. Este cenário vem crescendo a maneira na qual as pessoas se relacionam entre si e com o mundo, criando novos costumes e abrindo espaço a novas possibilidades para resolução dos problemas da vida cotidiana.

7.1 Contribuições

Com este trabalho é possível evidenciar a possibilidade de construir um sistema completo de rastreamento e controle do alarme de um veículo, bem como da trava elétrica e dos vidros elétricos do mesmo.

A construção dessa plataforma idealiza uma nova abordagem para atuar diante deste nicho de mercado que necessita cada vez mais de soluções específicas e que se adaptem a necessidade de cada usuário.

Esta plataforma de rastreamento, comunicação e controle de funcionalidades de um veículo descrita neste trabalho, é um exemplo de inovação no setor de rastreamento, pois devido à integração das tecnologias de hardware e software, as possibilidades de compreender novos mercados dentro deste segmento se tornou uma realidade alcançável.

O presente projeto apresentou uma forma rápida e eficiente de implantar serviços completos de localização, processamento de dados de rastreamento e controle de algumas funcionalidades do veículo.

7.2 Trabalhos Futuros

Embora o protótipo apresentado possa realizar as funções especificadas no decorrer do trabalho, o mesmo não poderá substituir o controle dos recursos do veículo através de um controle via sinais de rádio, devido ao fato de que a rede da operadora não possui cobertura em todos os lugares e o usuário poderá ser incapaz de utilizar seu veículo. No caso onde não haja cobertura da rede da operadora o sistema também deverá ficar inativo para impedir que alguma SMS seja enviada com conteúdo inválido.

Como a memória interna de um dispositivo móvel nem sempre é grande poderá ser feito um servidor para armazenar as informações referentes as posições do veículo e não armazená-las em um banco de dados no próprio *smartphone*.

Para aumentar ainda mais a segurança poderá ser implementado algoritmos de cadastramento de usuários e seus respectivos veículos no aplicativo Android e a implementação de algoritmos de criptografia para proteger as informações referentes ao veículo e ao usuário.

REFERÊNCIAS

ALVEZ, Uiliam N. L. T.; GEBARA, Munif, **Framework para supervisorio de sistema automatizado via celular**. VII EPCC (Encontro Internacional de Produção Científica Cesumar), Maringá – PR, out. 2011.

ANDROID. **Android, the worlds’ most popular mobile plataforma**. 2013a. Disponível em: <<http://developer.android.com/about/index.html>>. Acesso em: 22 mai. 2013.

ANDROID. **Exploring the SDK**. 2013b. Disponível em: <<http://developer.android.com/sdk/index.html> >. Acesso em: 22 mai. 2013.

ANDROID. **ADT Plugin**. 2013c. Disponível em: <<http://developer.android.com/tools/sdk/eclipse-adt.html>>. Acesso em: 13 jul. 2013.

ANDROID. **Google Maps Android API**. 2013d. Disponível em: <<http://developer.android.com/google/play-services/maps.html>>. Acesso em: 16 jun. 2013.

ANDROID. **Managing Virtual Devices**. 2013e. Disponível em: <<http://developer.android.com/tools/devices/index.html>>. Acesso em: 20 ago. 2013.

APPLE BRASIL, **IOS**. Disponível em: <<http://www.apple.com/br/ios/what-is/>>. Acesso em: 15 mai. 2013.

ARVUS, **Sistema de posicionamento global**. 2014. Disponível em: <http://www.arvus.com.br/publicacoes_exibe.html?id=2 >. Acesso em: 15 jun. 2014.

ASSIS, Paulo Ueiner Moreira. **Sistema de rastreamento de veículos para empresas de transporte utilizando navegação por satélite**. 2010. 121 f. Monografia (Engenharia de Computação) – UniCEUB (Centro Universitário de Brasília). Brasília – DF. 2010.

BMW, **My BMW remote APP**. Disponível em: <http://www.bmw.com/com/en/insights/technology/connecteddrive/2010/convenience/vehicle_management/my_bmw_remote_app_information.html>. Acesso em: 14 mai. 2013.

CANTO, Eduardo L. **Como funciona o GPS?**, Revista Ciências Naturais, Aprendendo com o Cotidiano, São Paulo, 2010, Disponível em: <<http://www.moderna.com.br/lumis/portal/file/fileDownload.jsp?fileId=8A7A83CB30D6852A0130D80AECF35B22>>. Acesso em: 04 jun. 2013.

CONSTANTINI, Ulisses. **Aplicação móvel de auto localização baseada na ferramenta google maps**. 2009. 42 f. Monografia (Especialização em Tecnologia Java). – UTFPR (Universidade Tecnológica Federal do Paraná – Câmpus Pato Branco). Pato Branco – PR. 2009.

COSTA, Albano R. **Sistema de monitoramento e rastreamento por GPS e GSM**. 2012. 35 f. Monografia (Engenharia de Computação) – Universidade Federal do Rio Grande do Norte. Natal – RN. 2012.

DENARDIN, Gustavo Weber. **Apostila microcontroladores**. Disponível em: <http://www.pessoal.utfpr.edu.br/gustavo/apostila_micro.pdf> Acesso em 10 mar. 2014.

DANA, Lorena. **Conheça a história do Android, o sistema operacional mobile do Google**. 2011. Disponível em: <<http://super.abril.com.br/galerias-fotos/conheca-historia-android-sistema-operacional-mobile-google-688822.shtml#4>>. Acesso em: 14 jun. 2014.

DEPARTAMENTO NACIONAL DE TRÂNSITO – DENATRAN, **Frota de veículos**. Disponível em: <<http://www.denatran.gov.br/frota.htm>>. Acesso em: 22 mai. 2013.

ECLIPSE, **About the Eclipse Foundation**. Disponível em: <<http://www.eclipse.org/org/>>. Acesso em: 09 ago. 2013.

EHLERS, M. **Geoinformatics and digital earth initiatives: a German perspective**, Osnabrueck – Alemanha, v. 1, n. 1, p. 2, mar. 2008.

ÉPOCA, **No Brasil, 14% da população brasileira já tem um smartphone**. Disponível em: <<http://epocanegocios.globo.com/Informacao/Resultados/noticia/2012/05/no-brasil-14-da-populacao-ja-tem-um-smartphone.html>>. Acesso em: 23 ago. 2013.

FERREIRA, Daniel Alves. **Sistema de alarme integrando GPS e celular**. 2010. 65 f. Monografia (Engenharia de Computação) – UniCEUB (Centro Universitário de Brasília). Brasília – DF. 2010.

G1, **Operadora lança G1, primeiro celular com a plataforma Google**. Disponível em: <<http://g1.globo.com/Noticias/Tecnologia/0,,MUL770366-6174,00-OPERADORA+LANCA+G+PRIMEIRO+CELULAR+COM+A+PLATAFORMA+GOOGLE.html>>. Acesso em: 21 jun. 2013.

Garmin, **What is GPS?**. Disponível em: <<http://www8.garmin.com/aboutGPS/>>. Acesso em: 22 jul. 2014.

HASEGAWA, Júlio, K.; GALO, Mauricio; MONICO, João, F. G.; IMAI, Nilton, N., **Sistema de localização e navegação apoiado por GPS**, XX Congresso Brasileiro de Cartografia, Recife – PE, 1999.

KLEINA, Nilton. **Android domina 79% do mercado de smartphones, mas crescimento diminui**. Disponível em: <<http://www.tecmundo.com.br/android/49757-android-domina-79-do-mercado-de-smartphones-mas-crescimento-diminui.htm>>. Acesso em: 20 jun. 2014.

L293, **Datasheet**. Disponível em:

<<http://pdf.datasheetcatalog.com/datasheet/stmicroelectronics/1328.pdf>>. Acesso em: 12 jul. 2014.

LEMOS, André; JOSGRILBERG, Fabio. **Comunicação e mobilidade: Aspectos Socioculturais das Tecnologias Móveis de Comunicação no Brasil**. Salvador: EDUFBA, 2009.

NETRASTER, **Net X, Controle remoto via celular**. Disponível em: <<http://netraster.com.br/index.php/solucoes-2/>>. Acesso em: 12 mai. 2013.

NMEA. **National marine electronics association**. (2008). Disponível em <<http://www.nmea.org/>>. Acesso em: 22 abr. 2014.

PERALTA, Bruno. **Primeira SMS enviada faz 20 anos amanhã**, Disponível em: <<http://www.tecnologia.com.pt/2012/12/primeira-sms-enviada-faz-20-anos-amanha/>>. Acesso em: 15 ago. 2013.

PIMENTEL, Fabiana. **Aplicativo da BMW que controla carro pelo celular é lançado para Android**, Revista InfoMoney, São Paulo, ago. 2012. Disponível em: <<http://www.infomoney.com.br/minhas-financas/carros/noticia/2512052>>. Acesso em: 20 mai. 2013.

PIR Sensor, **Datasheet**. Disponível em:

<<http://www.parallax.com/sites/default/files/downloads/555-28027-PIR-Sensor-Prodcut-Doc-v2.2.pdf>>. Acesso em: 02 jul. 2014.

PRESSMAN, R. **Engenharia de software**, 5ª ed., Rio de Janeiro: McGraw-Hill, 2002.

ROQUE, Cassiano G.; OLIVEIRA, Ivan C.; FIGUEIREDO, Priscila P.; BRUM, Everton V. P.; CAMARGO, Mairo F. **Georreferenciamento**. Alta Floresta – MT, v. 4, n. 1, p. 88-100, 2006.

SANTANA, Otávio G. **Por que Java?**, Disponível em: <<http://www.devmedia.com.br/por-que-java/20384>>. Acesso em: 23 ago. 2013.

SASCAR, **Sascar APP**. Disponível em: <http://www.sascar.com.br/faces/sub_prod_sascar_app_new_template>. Acesso em: 12 mai. 2013.

SILVA, Bruno L. R. **Sistema de controle do trio automotivo por meio de SMS**. 2012. 89 f. Monografia (Engenharia de Computação) – UniCEUB (Centro Universitário de Brasília). Brasília – DF. 2012.

SILVA, Edvaldo A. L. **Formatos narrativos para dispositivos móveis: estudo de caso do seriado transmidiático o castigo final**. 2010. 162 f. Dissertação de Pós-Graduação em Artes, música e multimeios – UNICAMP (Universidade Estadual de Campinas). São Paulo – SP. 2010.

SOMMERVILLE, Ian. **Engenharia de software**. 8. ed. São Paulo: Pearson, 2007.

SQLite, **About SQLite**. Disponível em: <<http://www.sqlite.org/about.html>>. Acesso em: 30 jul. 2013.

TANENBAUM, Andrew S. **Sistemas operacionais modernos**, 3ª ed., São Paulo: Pearson, 2009.

TEXAS INSTRUMENTS. **Tiva™ C Series TM4C123G launchpad evaluation board user's guide**. Disponível em: <<http://www.ti.com/lit/ug/spmu296/spmu296.pdf>>. Acesso em: 08 jul. 2014.

TOZETTO, Claudia. **Venda de smartphones com Android mais que dobra em um ano, diz estudo**. Disponível em: <<http://tecnologia.ig.com.br/2012-11-14/venda-de-smartphones-com-android-mais-que-dobra-em-um-ano-diz-estudo.html>>. Acesso em: 14 jun. 2014.

VANDRÉ, Carlos. Alô, aqui é seu carro. **Revista Quatro Rodas**, São Paulo, fev. 2007. Disponível em: <http://quatorrodas.abril.com.br/autoservico/cumpre/conteudo_215006.shtml>. Acesso em: 22 mai. 2013.

WIRING. **Wiring**. Disponível em: <<http://wiring.org.co/>>. Acesso em: 11 jun. 2014.

ZELENOVSKY, Ricardo; MENDONÇA, Alexandre. **Introdução aos sistemas embutidos**. Disponível em: <<http://www.mzeditora.com.br/artigos/embut.htm>> Acesso em 10 mar. 2014.

ZOOM, **O que é um celular Android?** Disponível em: <<http://www.zoom.com.br/celular/deumzoom/o-que-e-um-celular-android>>. Acesso em: 16 jun. 2014.