

MEC-SETEC
INSTITUTO FEDERAL MINAS GERAIS – *Campus Formiga*
Curso de Ciência da Computação

SISTEMA DE ANÁLISE DE CONSUMO DE COMBUSTÍVEL DE VEÍCULOS AUTOMOTORES

Rodrigo de Deus Almeida

Orientador: Prof. Me. Fernando Paim Lima

FORMIGA – MG

2017

Rodrigo de Deus Almeida

Sistema Móvel de Análise de Consumo de Combustível de Veículos Automotores

Trabalho de Conclusão de Curso apresentada
ao Instituto Federal de Educação, Ciência e
Tecnologia de Minas Gerais - Campus For-
miga, como requisito parcial para obtenção
do grau de Bacharel em Ciência da Compu-
tação.

Área de concentração: Computação.

Instituto Federal de Educação, Ciência e Tecnologia de Minas Gerais - Campus Formiga
Ciência da Computação

Orientador: Prof. MSc. Fernando Paim Lima

Formiga - MG

2017

Almeida, Rodrigo de Deus.
005.1 Sistema Móvel de Análise de Consumo de Combustível de Veículos
A434s Automotores / Rodrigo de Deus Almeida. -- Formiga : IFMG, 2017.
63p. : il.

Orientador: Prof. Msc. Fernando Paim Lima
Trabalho de Conclusão de Curso – Instituto Federal de Educação,
Ciência e Tecnologia de Minas Gerais – *Campus* Formiga.

1. Software - OBD II. 2. Android. 3. Consumo de Combustível.
4. Engenharia de Software. 5. Software. I. Título

CDD 005.1

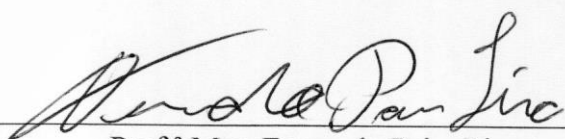
Rodrigo de Deus Almeida

Sistema móvel de análise de consumo de combustível de veículos automotores

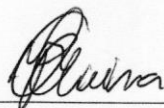
Trabalho de Conclusão de Curso apresentado ao
Instituto Federal de Minas Gerais-Campus Formiga,
como Requisito parcial para obtenção do título de
Bacharel em Ciência da Computação.

Aprovado em: 08 de junho de 2017.

BANCA EXAMINADORA



Prof.º Msc. Fernando Paim Lima
(Orientador)



Prof.ª Dra. Paloma Maira de Oliveira
(Avaliador 01)



Prof.º Dr. Bruno Ferreira
(Avaliador 02)

Resumo

O consumo desenfreado de combustíveis fósseis causa impactos tanto ambientais quanto econômicos. A redução de consumo de combustível pode ser realizada por meio da utilização de novas tecnologias e a recorrente manutenção dos componentes mecânicos, além da adoção de bons hábitos de condução do veículo. Um sistema capaz de auxiliar o condutor a realizar análises de gastos de combustível de seus veículos poderia propiciar menos desperdício, proporcionado por falhas que prejudiquem a eficiência de consumo. Este trabalho descreve a implementação de um sistema móvel Android que apoie o condutor no monitoramento e acompanhamento do consumo de combustível de seus veículos automotivos, proporcionado que o usuário detecte falhas e irregularidades que estejam influenciando no consumo habitual de combustível, para que sejam tomadas providências em relação a manutenção ou reeducação do condutor em hábitos específicos para economia de combustível. Para isso, a aplicação contará com a conexão com o sistema de diagnóstico embarcado veicular OBD-II através de um leitor *Bluetooth*. Os resultados obtidos mostraram-se promissores, além de um grande potencial em trabalhos futuros.

Palavras-chave: Software - OBD II. Android. Consumo de Combustível. Engenharia de Software. Software

Lista de abreviaturas e siglas

ABS	<i>Antilock Breaking System</i> (Sistema anti-bloqueio)
API	<i>Application Programming Interface</i> (Interface de Programação de Aplicativos)
CAN	<i>Controller Area Network</i> (Rede de Controle de Área)
CARB	<i>California Air Resource Board</i> (Conselho de Recursos Aéreos da Califórnia)
COM	<i>Communication Port</i> (Porta de Comunicação)
CONPET	Programa Nacional da Racionalização do Uso dos Derivados do Petróleo e Gás Natural
ECU	<i>Engine Control Unit</i> (Unidade de Controle do Motor)
EIA	<i>Energy Information Administration</i> (Administração de Informação Energética)
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
MAF	<i>Mass Air Flow</i> (Fluxo de Massa de Ar)
MIL	<i>Malfunction Indicator Lamp</i> (Lâmpada indicadora de mau funcionamento)
MPG	<i>Miles Per Gallon</i> (Milhas Por Galão)
OBD	<i>Onboard Diagnostic</i> (Diagnóstico de bordo)
PIN	<i>Personnal Identification Number</i> (Número de Identificação Pessoal)
PVA	Posição da Válvula de Aceleração
RPM	Rotações por Minuto
SIG	<i>Special Interest Group</i> (Grupo de Interesse Especial)
SQL	<i>Structured Query Language</i> (Linguagem de Consulta Estruturada)
TPS	<i>Throttle Position Sensor</i> (Sensor de posição do acelerador)
USB	<i>Universal Serial Bus</i> (Barramento Serial Universal)

Sumário

1	INTRODUÇÃO	11
1.1	Problematização	12
1.1.1	Trabalhos Relacionados	13
1.2	Solução Proposta	14
1.2.1	Objetivos	17
1.2.1.1	Objetivos Específicos	17
1.3	Estrutura do Trabalho	17
2	FUNDAMENTAÇÃO TEÓRICA	19
2.1	Consumo de Combustível	19
2.2	OBD-II	21
2.3	<i>Bluetooth</i>	23
2.4	Android	23
2.5	Android Studio	24
2.6	<i>SQLite</i>	25
2.7	<i>Android OBD Reader</i>	25
3	PROJETO	27
3.1	Definição do Escopo	27
3.2	Requisitos	29
3.2.1	Requisitos Funcionais	29
3.2.2	Requisitos Não Funcionais	31
3.2.3	Regra de Negócio	31
3.3	Banco de Dados do Sistema	31
3.4	Codificação	34
3.4.1	Receptor <i>Bluetooth</i>	34
3.4.1.1	Aplicação da Biblioteca <i>Android OBD Reader</i>	35
3.4.2	Calculador de Distância Percorrida	36
3.4.3	Calculador de Média da Posição da Válvula de Aceleração	37
3.4.4	Detector de Abastecimento	38
3.4.5	Máquina de Estados	39
3.4.6	Interface	41
3.5	Dificuldades Encontradas	43
4	RESULTADOS	45
4.1	Testes	45

4.1.1	Testes em Laboratório	45
4.1.2	Testes de Campo	47
4.2	Comportamento do Sistema no Teste de Campo	50
4.3	Verificação do Método	52
5	CONCLUSÃO	55
5.1	Trabalhos Futuros	55
5.1.1	Conceito Corporativo	56
5.1.2	Conceito Global	57
	REFERÊNCIAS	59
	ANEXOS	61
	ANEXO A – CÓDIGO FONTE E APLICATIVO	63

1 Introdução

Os meios de transporte têm um papel fundamental nas sociedades atuais, sendo responsáveis pela movimentação de bens e pessoas. O seu alto consumo energético, coloca os meios de transporte em um lugar de destaque no planejamento e gestão, principalmente pelo fato de que este setor possui uma forte concentração de veículos que possuem combustíveis derivados de petróleo, que além de possuírem reservatórios limitados e um alto preço no mercado, existe ainda uma questão ambiental com seu relacionada ao seu consumo abusivo. Portanto, qualquer otimização ou redução de sua utilização será sempre bem-vinda pela sociedade.

Uma tecnologia de análise do consumo de combustível em veículos automotores seria ideal para controlar, gerir e otimizar o custo dos abastecimentos, além de colaborar com a questão ambiental com o consumo sustentável e menor emissão de gases poluentes (FURRIELA, 2000), atendendo as necessidades de recursos naturais das presentes gerações, sem comprometer as necessidades das futuras gerações. Segundo Mattos (2001), o setor de transportes apresenta um enorme potencial de redução de emissões de CO_2 , principalmente em metrópoles, nas quais possuem uma grande frota de veículos em circulação diariamente.

Diante desta situação e preocupação, foi desenvolvido em 1980 o sistema de diagnóstico OBD (*On-Board Diagnostic*) para controle e monitoramento de emissões de gases veiculares, que reúne diversos dados providos por sensores distribuídos no veículo e os apresenta de forma padronizada através de uma porta serial, que pode ser acessada por diversos dispositivos específicos. Atualmente a versão é conhecida como OBD-II entre as montadoras e suas variantes são adotadas em vários países. Com uma variedade de leituras de dados sobre o veículo que podem ser coletados e interpretados por equipamentos e softwares específicos (TOMIOKA; SOUZA, 2016).

Este trabalho visa apresentar e descrever um protótipo de um sistema de informação gerencial (SIG) aliado à um dispositivo OBD-II que realizaria o acompanhamento do consumo de combustível de veículos automotores, já que o padrão oferece algumas informações em tempo real do veículo que podem ser processadas e armazenadas pelo sistema a fim de gerar dados estatísticos do consumo daquele veículo, proporcionado a detecção manual pelo usuário de possíveis defeitos, mal funcionamento mecânico ou até maus hábitos de condução do veículo que poderiam influenciar no aumento do consumo de combustível.

1.1 Problematização

De acordo com a EIA (*Energy Information Administration*), o consumo mundial de combustíveis fósseis deve continuar a crescer a um ritmo acelerado, com o aumento de 53% até 2035. Esse aumento contribui para o aumento do custo de aquisição desse tipo de combustível, já que se torna cada vez mais escasso.

O custo com combustíveis para uma empresa que possua uma frota de veículos é sempre motivo de preocupação para a maioria de seus gestores, pois, costumam representar uma grande parte das despesas gerais do negócio. Desta forma, qualquer medida que contribua para a eficiência energética será refletida em uma maior margem de lucro que o gestor poderá obter para sua empresa. Similarmente, um estudo realizado por [Arruda e Freitas \(2006\)](#) mostra que os gastos com transporte (combustível, taxi e ônibus, etc.) ocupam uma parcela de 12,31% nas despesas pessoais per capita da população brasileira.

Neste contexto, o setor de transportes tem uma grande relevância no aspecto ambiental já que segundo [Azuga \(2000\)](#), é responsável por aproximadamente 50% do consumo energético no Brasil e grande parte deste volume energético se deve aos combustíveis fósseis, que além de prover uma fonte não renovável de energia, emitem gases e partículas através da combustão, impactando na qualidade do ar, afetando a saúde das pessoas e contribuindo para o aumento do efeito estufa e do aquecimento global.

De acordo com o estudo realizado por [Chow et al. \(2012\)](#), as cidades de países em desenvolvimento têm níveis de poluição mais elevados que cidades em nações mais ricas. Apensar da grande utilização dos transportes públicos pelos seus habitantes, essas cidades não possuem a infraestrutura necessária para oferecer combustíveis limpos para a população, aliado à alta circulação de veículos antigos sem tecnologias que proporcionem a redução de emissão de gases na atmosfera.

Segundo [Arruda \(2014\)](#), a redução de consumo de combustível pode ser atingida por meio da adoção de novas tecnologias e a recorrente manutenção dos componentes mecânicos, pois existem diversos problemas de mal funcionamento mecânicos que influenciam negativamente na economia de combustível do veículo, como problemas na vela de ignição, sonda lambda, limpeza e substituição dos bicos injetores da injeção eletrônica, calibragem e balanceamento dos pneus e qualidade do óleo ([SOUZA, 2016](#)). Além disso, os hábitos de condução do veículo podem ser de grande influência em seu consumo de combustível total, como arrancadas bruscas, redução do acionamento da embreagem, utilização abusiva do equipamento de ar condicionado, constantes trocas de câmbios e entre outros.

O desenvolvimento de novas tecnologias para o controle do consumo de combustíveis e emissão de poluentes concentra-se nos laboratórios das montadoras e empresas automobilísticas ([ARRUDA, 2014](#)), porém muitas vezes o diagnóstico do consumo ex-

cessivo pode ser extremamente complexo, já que a análise do consumo envolve diversas variáveis que podem contribuir para seu aumento.

Por estes motivos, este trabalho vem propor uma ferramenta que auxilie o condutor a realizar análises de gasto de combustível de veículos automotores, de forma que permita que usuário detecte manualmente defeitos ou falhas que possam estar prejudicando a eficiência de consumo do mesmo.

1.1.1 Trabalhos Relacionados

No mercado de aplicativos para dispositivos móveis, existem diversas ferramentas que auxiliam o condutor a economizar combustível. Alguns aplicativos funcionam como uma calculadora de custos, na qual o usuário deve inserir manualmente algumas informações, como o preço do combustível ou quantidade de combustível, de modo que o sistema possa apresentar estratégias de redução de consumo como troca de tipo de combustível ou apresentarem dados estatísticos sobre o consumo do usuário. Exemplos de aplicativos com essa proposta são: *Carrorama*, *Econoflex (Android)*, *Gasolina ou Álcool*, *Chefia?*, entre outros.

Alguns aplicativos também oferecem o registro de abastecimento e consumo de veículos, como o *AutoCare*, *Carango (Android)*, *Carrorama* e *aCar*. O gerenciamento de consumo oferecido por estes sistemas depende da inserção manual de todas as informações dos abastecimentos realizados pelo veículo, podendo comprometer a integridade dos registros caso haja um descuido ou omissão do usuário, sendo capaz de prejudicar os resultados finais da análise de consumo.

Há também outros tipos de ferramentas capazes de se conectar com veículo via OBD-II, como o *Torque*, *OBD Car Doctor*, *Piston* e entre outros, que funcionam basicamente como computadores de bordo e analisadores de desempenho. Estas aplicações permitem o usuário acessar os vários sensores da ECU contidos no veículo em tempo real crítico, através de um dispositivo leitor OBD-II *Bluetooth* conectado ao automóvel. A vantagem deste tipo de abordagem é que as informações do veículo são capturadas passivamente, sem a necessidade da interação do usuário, proporcionando uma maior confiabilidade nos resultados de suas análises e relatórios. Porém, nenhum deles trata com detalhes a questão do consumo de combustível, exceto a ferramenta desenvolvido em [Fontes e Figueiredo \(2014\)](#): *Econodrive*.

O *Econodrive* trata-se de um sistema automotivo desenvolvido para sistemas móveis *iOS* capaz de coletar dados da ECU do veículo por meio do padrão OBD-II com propósito de realizar cálculos para determinar o consumo instantâneo e médio de combustível. O valor de consumo de combustível é obtido a partir dos valores de velocidade instantânea do veículo e a taxa de massa de ar nos cilindros do motor. O propósito desta

ferramenta é apenas apresentar o consumo instantâneo e a média de consumo do período em que o aplicativo foi iniciado, limitando a capacidade de análise e acompanhamento do consumo durante um longo período de tempo, além de não permitir o uso de mais um veículo ou uma comparação de consumo entre outros tipos de combustíveis. A tabela 1 resume os pontos positivos e negativos das categorias de ferramentas analisadas

Outra ferramenta interessante é o trabalho realizado em [Monteiro \(2013\)](#), que desenvolveu um protótipo de um sistema de monitoramento veicular tanto para frotas quanto para automóveis pessoais, utilizando também o sistema de diagnóstico OBD-II, que envia estes dados para um servidor via internet, utilizando uma placa microcontroladora programável *Arduino*. Deste modo, a ferramenta disponibiliza as informações sobre automóvel, possibilitando o monitoramento, inspeção e gerenciamento dos veículos pelo administrador do sistema.

Mesmo com as ferramentas disponíveis atualmente no mercado, ainda não há uma ferramenta específica que explore o monitoramento e acompanhamento do consumo de combustível através do OBD-II de forma totalmente precisa e consistente. A solução apresentada neste documento visa justamente explorar esta possibilidade oferecida pelo sistema de diagnóstico, utilizando recursos de baixo custo: Um dispositivo leitor *Bluetooth* OBD-II e um *smartphone* com sistema operacional Android.

Tabela 1 – Análise das ferramentas móveis disponível no mercado.

Ferramenta	Ponto Positivo	Ponto Negativo
<i>Econoflex</i>	Cálculo preciso	Inserção manual das informações do veículo
<i>AutoCare</i>	Acompanhamento do consumo de combustível	Inserção manual das informações do veículo
<i>Torque</i>	Conexão com veículo via OBD-II	Sem acompanhamento do consumo de combustível
<i>Econodrive</i>	Acompanhamento passivo do consumo de combustível	Imprecisão dos valores de consumo de combustível

Fonte: Autor.

1.2 Solução Proposta

Com base no cenário apresentado no capítulo 1.1, se faz necessário um sistema de informação gerencial que apoie o gestor de frotas de veículos (ou até o proprietário individual) no monitoramento e acompanhamento do consumo de combustível de seus veículos automotores, possibilitando que o usuário detecte por meio da observação, falhas e irregularidades de desempenho por meio de informações de apoio a decisão, para que sejam tomadas providências em relação a manutenção ou reeducação do condutor em hábitos específicos para economia de combustível.

Neste trabalho são descritos os processos de coleta e interpretação de dados do veículo em tempo real, cálculo de informações relevantes para o consumo de combustível de veículos automotivos e apresentação destas informações, através de uma plataforma móvel. Em geral, o protótipo possuirá o objetivo de apresentar informações sobre o consumo do veículo para o usuário, de modo a facilitar a detecção de valores divergentes aos valores naturais de consumo previsto para aquele veículo.

O processo de coleta dos dados do veículo será apoiado pela biblioteca de desenvolvimento *Android OBD Reader*, disponibilizada por Paulo Pires no site de compartilhamento de código e *software GitHub*. Após a coleta e interpretação dos dados do veículo, cabe ao sistema a responsabilidade de executar cinco tarefas fundamentais:

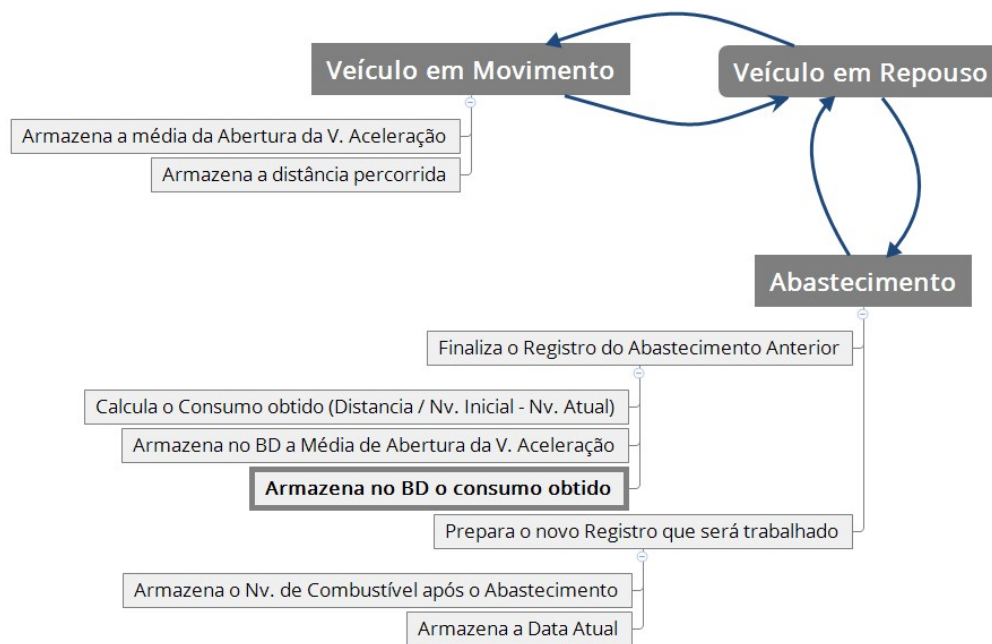
- Detectar se há um abastecimento de combustível ocorrendo naquele momento e a quantidade de combustível incrementada ao tanque de combustível;
- Armazenar a média da posição da válvula de aceleração (PVA) durante os períodos em que o veículo esteja em movimento;
- Armazenar a distância percorrida pelo veículo;
- Calcular e gravar a partir dos dados armazenados, o consumo de combustível obtido pelo veículo durante determinados períodos; e;
- Apresentar ao usuário os registros e relatórios gráficos do consumo de combustível atingido pelo veículo.

O período em que os dados de consumo de combustível são calculados e armazenados serão entre os abastecimentos detectados pelo sistema, assim como a sua média da PVA (Posição da Válvula de Aceleração), ou seja, para cada abastecimento realizado no veículo, haverá um registro de consumo de combustível associado a ele. No primeiro uso do sistema, o período de coleta parte da primeira conexão ao veículo até o momento do próximo abastecimento realizado. A Figura 1 ilustra o ciclo de vida da coleta das informações do veículo para o registro no banco de dados:

Os registros do consumo de combustível serão persistidos no banco de dados do aplicativo possibilitando serem consultados e apresentados ao usuário através de um relatório gráfico, como demonstrado pela Figura 2 que trata-se de uma captura de tela real do sistema desenvolvido neste trabalho.

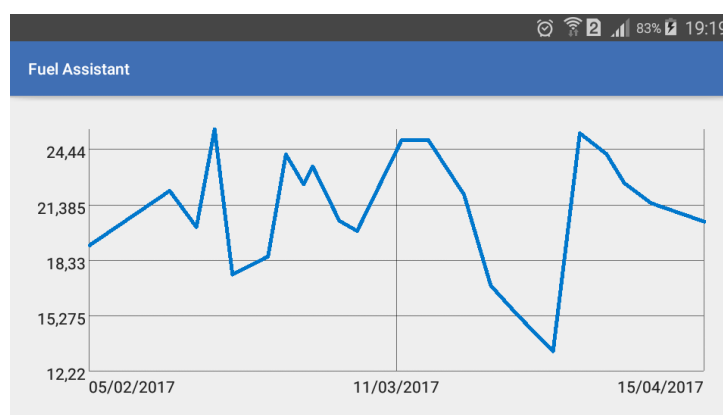
Com o resultado do relatório gráfico, o usuário poderá identificar oscilações no consumo de combustível de seu veículo, e tomar as devidas providências de reparação e manutenção com a finalidade de se obter o consumo esperado para aquele veículo.

Figura 1 – Ciclo de vida da coleta de dados do veículo



Fonte: Autor.

Figura 2 – Resultado do relatório gráfico de consumo de combustível apresentado pelo sistema, no qual o eixo X representa a data do registro e o eixo Y o consumo de combustível registrado



Fonte: Autor.

1.2.1 Objetivos

Desenvolver um sistema móvel para a plataforma Android capaz de receber e coletar informações do dispositivo leitor OBD-II instalado em um veículo automotor de forma totalmente passiva, e apresentar ao usuário as informações de consumo de combustível coletadas durante os percursos realizado por ele.

1.2.1.1 Objetivos Específicos

- Pesquisar e compreender detalhadamente o funcionamento do protocolo OBD-II;
- Estudar as ferramentas de desenvolvimento móvel no sistema Android;
- Pesquisar e analisar ferramentas que realizem a comunicação entre o leitor OBD-II e um sistema móvel Android;
- Pesquisar formas de se obter o consumo de combustível de veículos a partir de dados que podem ser obtidos através do OBD-II;
- Definir o escopo do sistema;
- Realização de testes de laboratório e de campo; e;
- Documentar o desenvolvimento e os resultados obtidos.

1.3 Estrutura do Trabalho

Este trabalho está dividido em seis capítulos. O capítulo 1 aborda o estado atual do problema, trabalhos relacionados, solução proposta e objetivos.

O capítulo 2 é responsável pela fundamentação teórica das ferramentas e tecnologias utilizadas no desenvolvimento do sistema. As tecnologias OBD-II, *Bluetooth*, Android e *SQLite* estão expostas de maneira sucinta, destacando principalmente os aspectos interessantes que envolvem o desenvolvimento e a codificação do projeto.

O capítulo 3 apresenta os requisitos funcionais e não funcionais e delimitação do escopo do sistema. Além do banco de dados do sistema e apresentação das tabelas que o compõe. As estratégias lógicas e funcionais de codificação do projeto, além de expor as dificuldades encontradas durante seu desenvolvimento.

O capítulo 4 apresenta os resultados dos testes realizados no sistema, algumas observações sobre o comportamento do sistema mediante a experimentação e a comparação destes resultados com outras alternativas de obtenção do consumo de combustível.

O capítulo 5.1 exclusivamente expõe os trabalhos e aplicações futuras que podem ser exploradas a partir do sistema desenvolvido neste trabalho, além de discussões de viabilidade destas concepções.

E finalmente o capítulo 5 possui as considerações finais sobre a idealização do projeto, execução e resultados obtidos.

2 Fundamentação Teórica

Nesta sessão estão abordados os temas relacionados ao entendimento e fundamentação necessária para o desenvolvimento do trabalho, desde os princípios de cálculos de consumo de combustível até as bibliotecas de desenvolvimento móvel utilizada na coleta das informações do veículo.

2.1 Consumo de Combustível

Quando nos referimos a consumo de combustível, podemos associar o combustível como a matéria prima consumida para obter um deslocamento medido como distância, ou seja, o objetivo da queima de combustível é fornecer trabalho para os eixos do veículo converter esta energia em locomoção. Portanto, o produto final desta conversão é a distância.

De acordo com [Silva \(2014\)](#), uma equação que trate o consumo baseado em um volume de combustível e distância percorrida pode ser representada pela Fórmula 2.1:

$$C = \frac{S_t}{V_t}$$

Consumo de combustível generalizado, tal que C representa o Consumo, S_t a distância total e V_t o total do volume de combustível consumido (2.1)

Uma das unidades de medida mais comum para o consumo de combustível no Brasil é representada por Km/L (quilômetros por litro), também referenciada como autonomia, que simplesmente associa a distância percorrida para cada unidade de volume litro, ou seja, do ponto de vista da economia de consumo, quanto maior o valor do consumo representado, maior é a economia do recurso de combustível.

A unidade de Quilômetros por litro não é utilizada internacionalmente. Cada região do mundo adota uma unidade, como nos EUA, a unidade mais conhecida é o MPG (milhas por galão), na Europa o padrão é L/100 km (litros por cem quilômetros), que também é adotada por diversos países sul-americanos. Porém utilizaremos somente a unidade Km/L durante o desenvolvimento deste trabalho.

Como atualmente os veículos são constantemente abastecidos em quantidades variadas, o cálculo do volume consumido para atingir uma determinada distância deve ser representado pela variação do volume de combustível durante o deslocamento, portanto, a equação ideal para volumes variantes é definido pela expansão da Fórmula 2.2:

$$C = \frac{S_t}{(V_i - V_f)}$$

Consumo de combustível expandida, tal que C representa o Consumo, S_t a distância total, V_i o volume inicial e V_f o volume final de combustível. (2.2)

Para determinação da distância, podemos utilizar uma equação que associa a distância percorrida à velocidade média e um intervalo de tempo, do mesmo modo que está definida em Gaspar (2009) e na Fórmula 2.3:

$$S = V_m * \Delta t$$

Equação da distância percorrida baseada na velocidade média e no intervalo de tempo necessário para percorre-la, tal que S representa a distância, V_m a velocidade média e Δt a variação de tempo. (2.3)

A variação de tempo Δt assim como volume de combustível na Fórmula 2.2, trata-se da diferença entre o tempo final e inicial do deslocamento, respeitando a compatibilidade de unidade com a velocidade (Velocidade em Km/h, tempo deve ser notado em horas, para m/s tempo deve ser notado em segundos, etc.)

$$\Delta t = t_f - t_i$$

Equação da variação de tempo, que simplesmente consiste na diferença entre o tempo final t_f no tempo inicial t_i . (2.4)

Normalmente, um veículo alterna do estado de movimento e repouso constantemente conforme sua utilização. Nos instantes em que se encontra em repouso, sua velocidade é obviamente igual a zero, portanto, para uma maior coerência e precisão no cálculo da distância percorrida, através da Fórmula 2.3, a distância total será calculada apenas nos instantes em que o veículo estiver em movimento (durante uma corrida). Desta forma, a distância total será a soma das distâncias obtidas por cada corrida individualmente. A equação de obtenção da distância total do veículo será definida pela Fórmula 2.5:

$$S_t = \sum_{i=1}^C S_i$$

Equação da distância total percorrida S_t , que consiste no somatório das distâncias individuais S_i obtidas por cada corrida C . (2.5)

Então, a equação completa do consumo de combustível veicular que será utilizada neste trabalho é definida pela expansão da Fórmula 2.2, com junção da Fórmula 2.5. Suas aplicações estão discutidas, justificadas e exemplificadas nas sessões de codificação deste documento.

$$C = \frac{\sum_{i=1}^C S_i}{(V_i - V_f)}$$

Equação expandida do consumo de combustível veicular (2.6)

Outro fator que influencia na análise de consumo de combustível, é a média da posição da válvula de aceleração (PVA), já que esta válvula é acionada pelos pedais de aceleração e são responsáveis por injetar combustível no motor do veículo, de forma que sua abertura mínima (0%) possui injeção nula de combustível, e sua abertura máxima (100%) injetará a quantidade máxima de combustível permitida pelo veículo (CAMPIGLIA, 2007).

O valor médio de abertura desta válvula durante a utilização do veículo pode variar por diversos motivos, como problemas mecânicos relacionados a injeção eletrônica e até maus hábitos de condução, como arrancadas bruscas (COSTA, 2002). A identificação de um valor médio elevado pode representar uma manutenção antecipada de um problema mecânico ou até uma medida corretiva aos hábitos de condução do motorista, prevenindo um possível desperdício de combustível ou um agravamento de um problema mecânico.

2.2 OBD-II

OBD (*Onboard Diagnostic*) trata-se de um sistema de diagnóstico embarcado veicular que permite o monitoramento e acesso a diversos componentes mecânicos e eletrônicos presentes no veículo por meio de uma interface serial, que atualmente está na sua segunda versão, OBD-II, que está padronizada em quase todo o mundo, e desde 2010 a legislação brasileira o adotou como obrigatória sua instalação para todos os veículos fabricados e comercializados no país.

Inicialmente, o projeto do OBD surgiu com a preocupação de conhecer e controlar os níveis de emissões dos veículos e adaptá-los aos padrões exigidos pelas legislações, Segundo Santana (2009), a primeira versão do OBD foi introduzida pelo CARB (*California Air Resource Board*) durante os anos 70, aplicada inicialmente somente no estado da Califórnia, porém, logo o governo dos Estados Unidos aplicou esta regulamentação em todo país, forçando a indústria automotiva a investir e aprimorar os diagnósticos *on-board* dos veículos (CERQUEIRA et al., 2009). Suas primeiras versões tinham também a responsabilidade de acionar uma luz indicadora instalada no painel do veículo quando algum problema mecânico era detectando, alertando o condutor sobre a anormalidade do sistema mecânico, esta funcionalidade é conhecida atualmente como sinalização de mau funcionamento ou MIL (*Malfunction Indicator Lamp*).

Como cada fabricante era livre para incorporar seus próprios conectores e funcionalidades dos sistemas de diagnóstico dos seus veículos produzidos, diversas outras versões

do OBD foram desenvolvidas, criando um problema: A falta de padronização nas informações fornecidas e conectores dos instrumentos de acesso a elas. Este problema tornou o acesso exclusivo para ferramentas proprietárias de cada fabricante. Até que nos anos 90 surgiu o padrão OBD-II (*On Board Diagnostic Second Generation*), que trouxe além de uma grande ampliação nos itens disponíveis para acesso e uma maior taxa de transmissão de dados, a padronização dos protocolos e métodos de conexão entre todas as montadoras (MACHADO; OLIVEIRA, 2007).

Após a padronização, o sistema de diagnóstico OBD-II passou a ser implantado em todos os veículos de passageiros e veículos comerciais leves produzidos no mundo todo. O órgão responsável pela regulamentação ainda é o CARB, no qual publica suas novas resoluções através de seu *website* ¹, que já conta com centenas de protocolos de informações de condições de funcionamento e sensores específicos distribuídos pelo veículo, como velocidade, rotações por minuto do motor, volume de combustível, temperatura do óleo do motor, nível emissão de gases poluentes, situação do freio ABS (*Antilock Breaking System*), posição da válvula de aceleração, taxa de massa de nos cilindros do motor entre outros. Porém, nem todos os veículos possuem todas estas opções implementadas, já que somente o sensor de nível de emissão de gases poluentes é obrigatório a instalação. No entanto, as montadoras atualmente estão buscando a modernização e aperfeiçoamento de seus sistemas veiculares e a cada ano vêm trazendo mais opções de leitura de dados OBD-II.

A coleta dos dados fornecidos pelo sistema OBD-II não prevê uma comunicação direta com um sistema computacional, fazendo necessário a utilização de dispositivos leitores conhecidos como *ELM327*, projetados para agir como intermediário entre a porta OBD-II e um sistema por meio de diversos tipos de protocolos de comunicação populares, como *Wi-Fi* (IEEE 802.11), *USB*, *Bluetooth* (IEEE 802.15.1) e entre outros (FONTES; FIGUEIREDO, 2014).

Existem no mercado diversas opções de modelos e fabricantes destes dispositivos. A Figura 3 demonstra alguns exemplos de leitores OBD-II (também conhecidos com *dongle*), o leitor com transmissão via *USB*, no qual em setembro de 2016, possui o preço de mercado próximo a R\$ 56,00 e o leitor com transmissão via *Bluetooth*, com preço de mercado aproximado de R\$ 28,00 em maio de 2017:

A escolha do sistema OBD-II para realização deste trabalho se justifica pelo fato da ferramenta atender os requisitos necessários para seu desenvolvimento, que são: coleta dos valores de nível de combustível, velocidade e posição da válvula de aceleração do veículo em tempo real, compatibilidade de comunicação com sistema móvel (*Bluetooth*) e baixo custo de aquisição de equipamentos (leitor OBD-II).

¹ <https://www.arb.ca.gov/msprog/obdprog/obdregs.htm>

Figura 3 – Adaptadores OBD a) Transmissão via USB b) Transmissão Via Bluetooth



Fonte: <https://www.amazon.com>

2.3 Bluetooth

Bluetooth é uma tecnologia para comunicação remota de dispositivos eletrônicos por meio de radiofrequência, caracterizado por seu baixo alcance, baixo consumo de energia, baixo custo e com taxa de transmissão relativamente altas em suas versões mais recentes. De acordo com Billo (2003), o padrão *Bluetooth* foi desenvolvido pelo SIG (*Special Interest Group*), um grupo formado por algumas companhias líderes mundiais nas telecomunicações, computação e indústrias de redes padronizado pelo IEEE como IEEE 802.15.1.

O *Bluetooth* é muito utilizado como forma de comunicação entre dispositivos. Sua mobilidade e facilidade motivou a escolha desta opção de transmissão para o desenvolvimento deste trabalho. É necessário a realização de um pareamento entre os dispositivos antes de estabelecer uma conexão, que é o caso dos leitores OBD-II e dispositivos Android, na qual é realizada pelo método tradicional de pareamento da versão do Android utilizada. Após o pareamento, o leitor estará preparado para enviar e receber os dados do veículo remotamente para o *smarthphone*. Mais detalhes em relação ao processo de coleta dos dados via *Bluetooth* estão apresentados no capítulo 3, no qual apresenta o procedimento de conexão e coleta dos pacotes enviados pelo leitor.

2.4 Android

O Android é o sistema operacional móvel multiplataforma desenvolvido pela empresa de tecnologia Google, que atualmente é líder mundial no mercado de sistemas móveis. O Android está presente e integrado à diversas plataformas como *smarthphones*, *tablets*, TV, *smarthwatches*, *smarthglasses* e até mesmo em computadores de bordo de veículos.

O Android é a primeira plataforma para aplicações móveis completamente livre e de código aberto (*Open source*), o que representa uma grande vantagem competitiva para sua evolução, uma vez que diversas empresas e desenvolvedores do mundo podem contribuir

para melhorar a plataforma. Isso torna o Android muito vantajoso para fabricantes de *smarthphones*, pois lhe permitem realizar alterações no código-fonte para criar produtos customizados sem precisar compartilhar as alterações com ninguém, e ainda de forma gratuita.

O sistema operacional Android é baseado no *kernel* do Linux, que é responsável por gerenciar a memória, os processos, *threads*, segurança dos arquivos e pastas, além de redes e *drivers* (LECHETA, 2016). Cada aplicativo no Android possui um processo no sistema operacional. Diversos processos e aplicativos podem ser executados simultaneamente, e o *kernel* do sistema operacional é responsável por realizar todo o controle de memória.

A quantidade de ferramentas e materiais para estudo e desenvolvimento oferecida por esta plataforma e disponibilizadas pela internet foi um dos principais motivos de sua escolha para construção deste trabalho, além do fato de que ela está totalmente consolidada no mercado e presente na grande maioria dos dispositivos móveis pelo mundo.

2.5 Android Studio

Android Studio é a IDE oficial de desenvolvimento para Android. O *Android Studio* foi anunciado no Google I/O 2013 e é baseado no *IntelliK IDEA* da *JetBrains* (LECHETA, 2016). O Android Studio apresenta grandes diferenças importantes se comparado ao Eclipse, que era a antiga ferramenta oficial de desenvolvimento.

Uma das principais diferenças entre o Eclipse e o Android Studio é o processo de compilação dos projetos. No Eclipse cada projeto é compilado do jeito clássico, como qualquer projeto Java dentro do Eclipse, sendo necessário a utilização de arquivos JAR para adicionar qualquer dependência ao sistema, como bibliotecas personalizadas. Porém no Android Studio, a compilação é feita pelo *Gradle*, que possui elementos de flexibilidade, gerenciamento de dependência e convenções eficazes.

Como se trata de uma ferramenta oficial do sistema operacional, a quantidade de materiais para estudo e pesquisa é superior em relação a outras ferramentas de desenvolvimento, além de possuir uma grande quantidade de ferramentas, facilidades e vantagens que auxiliam a construção do software, como por exemplo o gerenciador de dependências *Gradle*, que ao contrário de seu popular concorrente *Eclipse*, evita a inserção manual de dependência por meio de arquivos *.JAR*. Isso motivou a escolha desta ferramenta para o desenvolvimento do sistema deste trabalho além de ser uma ferramenta totalmente gratuita.

2.6 SQLite

SQLite trata-se de uma biblioteca de banco de dados SQL integrada ao Android. Segundo Oliveira (2014), o *SQLite* se destaca por possuir uma fácil configuração, gratuito e com código fonte aberto (*open source*) e não possuir um sistema de administração multiusuário, ou seja, somente a aplicação origem tem acesso aos dados contidos no banco. Uma característica desta biblioteca é que a leitura e escrita dos dados são realizadas diretamente em um único arquivo.

De acordo com a documentação da biblioteca no site oficial do *SQLite*², o banco de dados não é indicado para aplicações que envolvem um grande número de acessos simultâneos (alta concorrência), grandes conjuntos de dados global superiores a 140 *terabytes* ou o acesso de aplicações independentes. Mas para o armazenamento local do dispositivo, com baixa concorrência e com conteúdo inferior a 1 (um) *terabyte*, o *SQLite* será uma excelente solução. Portanto para o sistema desenvolvido neste trabalho, esta solução se adapta perfeitamente.

2.7 Android OBD Reader

Android OBD Reader trata-se de uma popular biblioteca API Java desenvolvida por Paulo Pires, capaz de receber pacotes de dados de um leitor *OBDII* ELM 327 de modo que o desenvolvedor tenha acesso a todas as informações fornecidas pelo dispositivo sem a necessidade de aprofundamento de conhecimentos do protocolo de transporte e interpretação dos pacotes de dados, direcionando os esforços de desenvolvimento para a aplicação em si. Esta biblioteca está disponível no site de compartilhamento de projetos *GitHub*³ onde conta com contribuições de vários desenvolvedores do mundo todo.

A escolha desta biblioteca se deve a facilidade de implementação e simplicidade no uso dos componentes OBD-II, que é realizada por uma conexão *Bluetooth* usual, tornando o desenvolvimento do trabalho um pouco mais ágil, já que não foi necessário o aprofundamento do protocolo de comunicação OBD-II *Bluetooth* e nem na compatibilidade entre os diversos tipos e modelos de leitores OBD-II, além da total compatibilidade com qualquer versão do Android presente no aparelho.

² <https://www.sqlite.org/docs.html>

³ <https://github.com/pires/android-obd-reader>

3 Projeto

Neste capítulo é documentada a modelagem do sistema desenvolvido no decorrer do projeto, o analisador móvel de consumo de combustível de veículos automotores, com premissas e objetivos definidos conforme o capítulo 1, levando em consideração a definição do escopo, a análise de requisitos, diagrama lógico do banco de dados, codificação, diagrama de telas e testes do sistema.

Os requisitos mínimos para o funcionamento do analisador móvel de consumo são:

- Dispositivo Android com versão a partir da 4.4 (*KitKat*)
- Leitor OBD-II *Bluetooth* ELM-327
- Veículo automotor com entrada serial OBD-II e que ofereça suporte e conexão com a rede CAN para:
 - Sensor de velocidade
 - Sensor de abertura da válvula de aceleração
 - Sensor de nível de combustível

3.1 Definição do Escopo

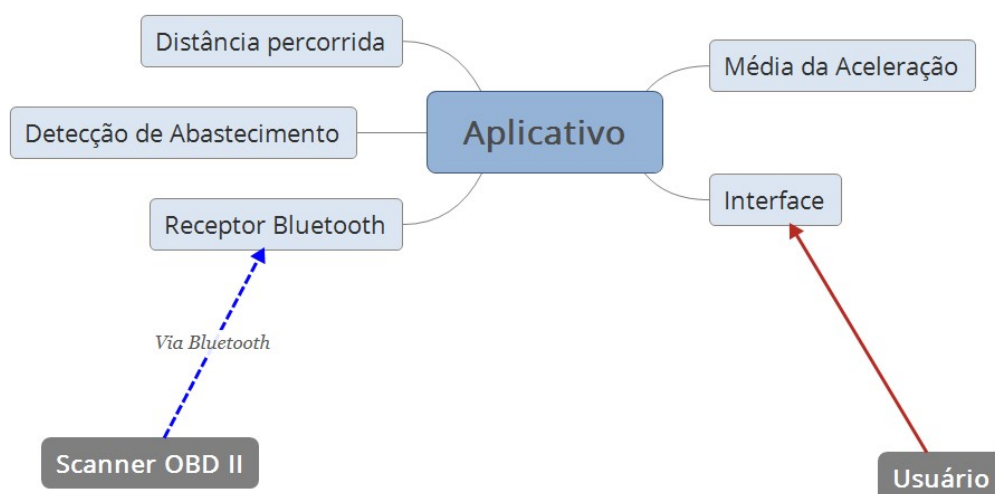
Para facilitar a compreensão e codificação do projeto, o sistema foi dividido em cinco módulos, cada um responsável por determinadas funções, operação e comportamento para que o sistema opere de forma adequada. Abaixo estão listados os módulos em que o sistema de acompanhamento de consumo de combustível deste trabalho utiliza, que estão especificados detalhadamente no capítulo 4 Codificação.

- **Receptor Bluetooth:** Este módulo é responsável por estabelecer uma conexão com o dispositivo leitor OBD-II *Bluetooth*, receber os pacotes com os dados providos pelo leitor e extrair as informações relevantes para o contexto do sistema;
- **Distância Percorrida pelo Veículo:** Uma informação significativa para análise do consumo do veículo é a distância percorrida por ele utilizando determinada quantidade de combustível. Este módulo é responsável pelo cálculo desta distância percorrida. A cada corrida realizada pelo veículo, uma nova distância é calculada e acrescentada no registro da distância total percorrida;

- **Média de Aceleração do Veículo:** Quando o dispositivo leitor está devidamente conectado ao sistema e o veículo se encontra em movimento, este módulo armazena os valores obtidos pelo sensor de aceleração do veículo, também conhecido por TPS (*Throttle Position Sensor*, ou sensor da posição da válvula borboleta). O valor instantâneo da posição da válvula é obtido periodicamente enquanto o veículo estiver com em velocidade de movimento superior a 3 Km/h e é realizada a média total destes valores armazenados;
- **Deteção de Abastecimento:** Este módulo se responsabiliza por monitorar o nível de combustível, para que seja feita a detecção de um abastecimento em tempo real crítico e armazenado o valor volumétrico daquele abastecimento. Sempre após um novo abastecimento, as informações sobre abastecimento anterior são gravadas no banco de dados da aplicação para a construção de registros e relatórios de consumo;
- **Interface Interativa:** Este módulo representa a interface do aplicativo, na qual o usuário irá interagir com o sistema, seja para configurações diversas, cadastro de veículos, conectar com o leitor OBDII, visualizar algumas informações do veículo em tempo real e visualizações dos relatórios e registros.

A essência deste projeto trata-se da interação e comunicação destes módulos simultaneamente de forma que o sistema trabalhe de forma adequada. A Figura 4 ilustra a disposição e integração dos módulos ao aplicativo.

Figura 4 – Ilustração da composição dos módulos na aplicação



Fonte: Autor.

3.2 Requisitos

Para apresentar o modo que o sistema funcionará, os próximos tópicos mostram respectivamente os requisitos funcionais, não funcionais e as regras de negócio que delimitam o seu escopo.

3.2.1 Requisitos Funcionais

No contexto do sistema, há cinco módulos responsáveis por tarefas exclusivas que trabalham simultaneamente definidos na sessão 6.1: Receptor *Bluetooth*, Calculador de Média de Aceleração, Calculador de Distância Percorrida, Detecção de Abastecimento e Interface. A seguir estão listados os Requisitos Funcionais de cada módulo:

Receptor Bluetooth

- RF01** O sistema deve ser capaz de ativar o receptor Bluetooth do aparelho;
- RF02** sistema deve ser capaz de fornecer todos os dispositivos Bluetooth pareados com o aparelho;
- RF03** O sistema deve ser capaz de estabelecer uma conexão via Bluetooth com um dispositivo leitor OBD-II;
- RF04** O sistema deve ser capaz de receber os pacotes enviados pelo leitor OBD-II e interpretá-los corretamente no contexto da aplicação;
- RF05** O sistema deve ser capaz de acessar os valores instantâneos de velocidade do veículo, nível de combustível e posição da válvula de aceleração do veículo;

Calculador de Distância Percorrida

- RF06** O sistema deve ser capaz de determinar quando o veículo está em velocidade superior a 3 Km/h;
- RF07** O sistema deve ser capaz de determinar a distância percorrida pelo veículo enquanto o aplicativo estiver em funcionamento;

Calculador de Média de Aceleração

- RF08** O sistema deve ser capaz de armazenar os valores instantâneos da abertura da válvula de aceleração do veículo, desde que a velocidade do veículo seja superior a 3 Km/h;

RF09 O sistema deve ser capaz de armazenar a média dos valores obtidos da abertura da válvula de aceleração do veículo entre os períodos de abastecimentos;

RF10 O sistema deve ser capaz de determinar a distância percorrida pelo veículo enquanto o aplicativo estiver em funcionamento;

Detecção de Abastecimento

RF11 O sistema deve ser capaz de monitorar o nível de combustível do veículo conectado;

RF12 O sistema deve ser capaz de detectar quando o nível de combustível sofrer um aumento e notificar que há um abastecimento sendo realizado no instante;

RF13 O sistema deve ser capaz de calcular o volume de combustível acrescentado no momento do abastecimento;

RF14 O sistema deve persistir as informações do abastecimento e consumo do veículo no momento em que um novo abastecimento é realizado;

RF15 O sistema deve informar por meio de notificação quando um abastecimento é realizado;

Interface

RF16 O sistema deve permitir a seleção e conexão com um dispositivo Bluetooth;

RF17 O sistema deverá informar que a conexão com o dispositivo Bluetooth está funcionando corretamente;

RF18 O sistema deve permitir a seleção do veículo cadastrado em que o leitor está conectado;

RF19 O sistema deve permitir alterar as informações do veículo, como marca, modelo, tipo de combustível e ano de fabricação;

RF20 O sistema deve permitir o cadastro de novos veículos no sistema;

RF21 O sistema deve permitir que o usuário defina se deseja receber notificações do aplicativo;

RF20 O sistema deve permitir a visualização dos registros de consumo e média de aceleração armazenados por veículo durante sua utilização; e

RF21 O sistema deve permitir a visualização gráfica dos registros armazenados por veículo, personalizando os resultados por períodos específicos (semanal, mensal, anual, etc.) e por veículos;

3.2.2 Requisitos Não Funcionais

RNF01 O sistema deve ser desenvolvido em plataforma móvel;

RNF02 O sistema deve ser desenvolvido na linguagem JAVA;

RNF03 O sistema deve utilizar o Banco de Dados SQLite; e

RNF04 O sistema deve trabalhar seus módulos de forma passiva e em **background** no dispositivo móvel (exceto o módulo de interface);

3.2.3 Regra de Negócio

RN01 O sistema deverá apresentar os registros e relatórios somente do veículo atribuído pelas configurações do aplicativo;

RN02 O sistema deverá apresentar os registros e relatórios somente do usuário atribuído pelas configurações do aplicativo;

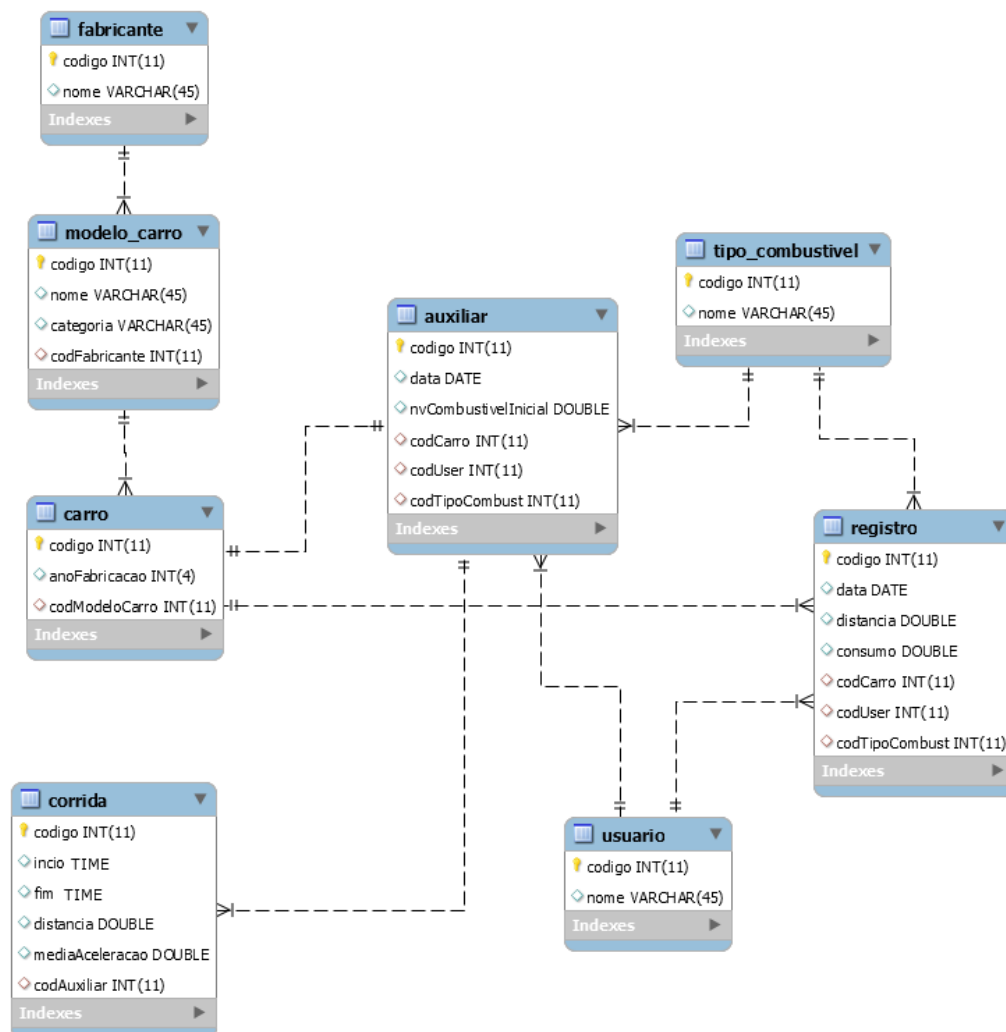
3.3 Banco de Dados do Sistema

Para possibilitar a apresentação das informações de consumo de combustível para o usuário do sistema, se faz necessário a implementação de um banco de dados capaz de armazenar as informações do veículo analisado e dados coletados sobre o consumo do veículo. As informações sobre o usuário não são fundamentais para o funcionamento deste sistema, porém sua implementação visa uma próxima expansão do sistema, que estão expostas no capítulo 5.1 Trabalhos Futuros.

O modelo lógico do banco de dados é a representação lógica de como o banco de dados está disposto no sistema. O sistema de gerenciamento de banco de dados escolhido foi o SQLite, porém o diagrama gerado para a concepção da Figura 5 foi construído no MySQL Workbench. Ele é responsável por exibir as tabelas, seus relacionamentos e suas respectivas cardinalidades, possibilitando o desenvolvimento do sistema a partir dele. A Figura 5 ilustra o diagrama do modelo lógico do banco de dados do sistema deste trabalho.

Como este sistema tem a premissa de analisar os dados coletados do veículo e compara-los entre diversas variáveis e situações, a tabela *Registro* possui um grande número de informações e chaves estrangeiras, podendo ser atribuída como a principal tabela do banco de dados. A intenção do sistema é coletar um grande número de registros durante a sua execução, que pode envolver diversos veículos com diversos tipos de combustível, possibilitando a personalização de consultas a estes registros, oferecendo ao usuário diversos tipos de informações para análise de consumo de combustível.

Figura 5 – Diagrama lógico do banco de dados do sistema



Fonte: Autor.

A tabela *fabricante* representa o fabricante ou montadora responsável pelo modelo do veículo (Ex.: Fiat, Ford, Volkswagen, etc).

A tabela *modelo_carro* representa o modelo do veículo automotor, com informações sobre sua categoria e fabricante (Ex.: Palio, Hatch, Fiat; Fusion, Sedan, Ford; Gol, Hatch, Volkswagen).

Os veículos cadastrados no sistema pelo usuário serão representados pela tabela *carro*, na qual possui as informações sobre o modelo do veículo e seu ano de fabricação (Ex.: (Palio, Hatch, Fiat), 2016). Os registros de consumo de combustível serão associados ao veículo por meio desta tabela. Desta forma, o usuário deterá o acesso as informações de consumo individualmente de cada veículo cadastrado.

As tabelas *usuario* e *tipo_combustivel* representam respectivamente a identificação

do condutor ou usuário do aplicativo e os tipos de combustíveis disponíveis no mercado (Ex. usuário: João, Bruno, Letícia; Ex. Tipo de combustível: Gasolina, Álcool, Dísel).

Durante o período entre um abastecimento e outro, são necessários o armazenamento de alguns dados sobre o presente abastecimento e as informações sobre o veículo, usuário e tipo de combustível, até que seja realizado o próximo abastecimento. Para isso existe a tabela *auxiliar*, que poderia ser denominada como um “Pré-Registro” ou registro temporário, ou seja, ela é responsável por armazenar as informações temporárias do registro, como a distância percorrida até o momento, nível de combustível após o abastecimento, data do abastecimento e as referências para acesso ao veículo, usuário e tipo de combustível (Ex.: (Palio, Hatch, Fiat), (João), (Gasolina), 12/12/2012, 60%). O relacionamento de *carro* e *auxiliar* é de 1 (um) para 1 (um), desta forma, existirá apenas um único registro temporário para cada veículo cadastrado no sistema.

Conforme foi discutido no escopo do sistema em 3.1, as informações sobre as corridas realizadas pelo veículo serão armazenadas no sistema por meio da tabela *corrida*. Nela serão armazenadas as informações sobre as corridas que ocorrem durante o período no qual a tabela de uso temporário *auxiliar* atua. Deste modo, para cada registro da tabela *auxiliar*, haverá um conjunto de corridas, logo, a distância total percorrida durante um registro será a soma das distâncias individuais de cada corrida pertencente a ela, e a média da PVA será a média das médias coletadas durante as corridas deste registro.

Finalmente, a tabela *registro* é a tabela central do sistema, representando o resultado de todos seus esforços. Ela é responsável por armazenar as informações completas sobre o consumo de combustível: Data do registro, distância percorrida, consumo de combustível e referências para acesso as informações do veículo, usuário e tipo de combustível.

No momento do início de um novo abastecimento, a tabela *auxiliar* é convertida para uma tabela *registro*, calculando o consumo de combustível no momento (a razão entre distância percorrida por todas corridas (S) e a diferença do nível de combustível inicial com o nível de combustível antes do atual abastecimento ($V_i - V_f$), do mesmo modo da Fórmula 3.1).

Após o termino do abastecimento e persistência da nova instância da tabela *registro*, a tabela *auxiliar* é preparada para iniciar uma nova captura de informações de consumo: Os dados contidos na tabela *auxiliar* são apagados, junto com todas as corridas pertencentes a ela, a data é alterada para a data atual e o nível de combustível inicial é alterado para o nível de combustível presente atualmente no veículo após o abastecimento.

3.4 Codificação

O aplicativo foi inteiramente desenvolvido no ambiente de desenvolvimento integrado (IDE) *Android Studio*, utilizando principalmente as bibliotecas padrões de compatibilidade recomendadas pelo Google, porém várias outras bibliotecas de customização de interface e de comunicação *Bluetooth*.

Conforme foi descrito na sessão 3.1 de escopo, o sistema é composto por cinco módulos de operação que trabalham de forma simultânea e são essenciais para o funcionamento do sistema. Neste capítulo, abordaremos os detalhes da codificação de cada módulo separadamente, justificando cada decisão tomada para seu desenvolvimento. Além de apresentar em detalhes a máquina de estados, que modela e coordena todas operações dos módulos.

Os módulos que compõe o sistema são convocados conforme o estado em que se encontra o veículo, exceto o módulo responsável pela interface com usuário. Na Figura 6 podemos observar esta distribuição em interação com o estado em que se encontra o veículo: Desconectado, Parado, em Movimento ou em Abastecimento.

O estado Parado se caracteriza quando veículo se encontra com velocidade inferior a 3 Km/h, o estado em Movimento permanece enquanto a velocidade se torna superior a 3 Km/h, e o estado em Abastecimento se inicia quando o nível de combustível presente no tanque de combustível atinge um valor maior em relação ao valor da última verificação (valores crescentes) e termina quando o nível se estabiliza em um mesmo valor durante 10 segundos.

A explanação mais detalhada destas etapas de mudança de estado e a justificativa dos valores de parâmetro estão demonstrados nas próximas sessões:

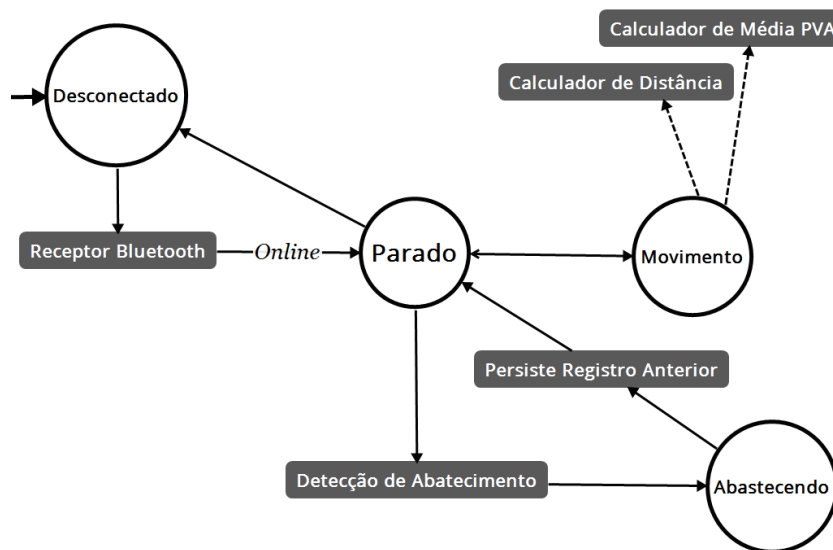
3.4.1 Receptor *Bluetooth*

O módulo receptor *Bluetooth* é uma ferramenta fundamental para o funcionamento do sistema, pois ele é responsável por capturar as informações do veículo providas pelo leitor OBD-II (também conhecidos como *Dongle*) e disponibiliza-las para os demais módulos. Desta forma, os demais módulos (exceto a Interface) somente serão autorizados a iniciar seus funcionamentos após a conexão do módulo de Recepção seja estabelecida com sucesso.

Primeiramente, para estabelecer uma conexão entre os dois dispositivos, é necessário o pareamento prévio entre eles, utilizando o método tradicional de pareamento para a respectiva versão do Android instalada no sistema.

Os modelos mais populares de leitores OBD-II possuem a senha PIN de pareamento padrão “1234”, porém pode ser distinta, dependendo do fabricante e modelo do leitor.

Figura 6 – Máquina de estados do veículo (Círculos), a distribuição dos módulos do sistema (Retângulos) e suas interações



Fonte: Autor.

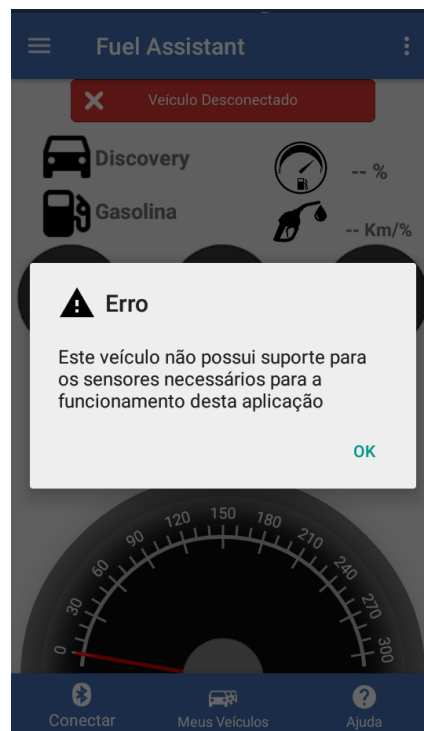
Após o pareamento entre os dispositivos, o sistema oferece o processo de conexão com o leitor OBD-II, através da classe *BluetoothAdapter* do Android, que representa a abertura para todas interações *Bluetooth*, tendo acesso a todos os dispositivos pareados e instanciar uma conexão entre eles.

3.4.1.1 Aplicação da Biblioteca *Android OBD Reader*

Com a conexão estabelecida, é possível executar os métodos de acesso as informações OBD-II do leitor através da biblioteca *Android OBD Reader* citada no capítulo 2.7. O primeiro passo é verificar se o veículo e o leitor atendem as necessidades do sistema, que é a presença do sensor de nível de combustível, sensor de velocidade do veículo e sensor abertura da válvula de aceleração.

Caso algum destes requisitos não estejam presentes no veículo, o processo de verificação de consumo de combustível do sistema não pode ser realizado, e é informado ao usuário qual requisito faltou para cumprir com as exigências mínimas para funcionamento, conforme ilustrado na Figura 7. Caso contrário, as informações dos sensores do veículo serão armazenadas e atualizadas a cada segundo, sendo oferecidas aos outros módulos durante toda a conexão.

Figura 7 – Mensagem de erro informada ao usuário quando um veículo não compatível com sistema é conectado



Fonte: Autor.

3.4.2 Calculador de Distância Percorrida

Quando o leitor OBD está devidamente conectado ao sistema, é possível verificar qual o estado atual do veículo: *Parado*, *Movimento* ou *Abastecendo*. O módulo Calculador de Distância Percorrida é responsável por fazer a transição do estado de *Parado* para o de *Movimento* e vice-versa, além de determinar a distância percorrida durante essa mudança de estado.

A cada segundo, após a atualização dos valores dos sensores do veículo pelo módulo de recepção *Bluetooth*, é realizada a verificação do valor da velocidade do veículo. Se este valor for superior a 3 Km/h, então o veículo passa para o estado de *Movimento*. Iremos nos referir ao período em que o veículo permanecer em movimento como a *Corrida*.

Se obter o valor da distância total de uma corrida é indispensável para se obter o consumo de combustível. Seguindo a Fórmula 3.1, o consumo será obtido através da razão entre a distância total percorrida e o volume de combustível consumido.

O valor de 3 Km/h foi estimado ao examinar valores nos quais não poderiam ser altos demais para deixar de capturar um deslocamento de uma distância consideravelmente relevante, reduzindo a precisão do cálculo de consumo ao final, e não tão baixo a ponto de classificar pequenas manobras realizadas pelo veículo como uma corrida, sobre-

carregando o sistema com informações irrelevantes. O valor de 3 Km/h foi adotado pois por ser menor que velocidade média de deslocamento da caminhada humana (5 Km/h), e é um valor suficientemente capaz de atender as exigências, já que um veículo que esteja abaixo desta velocidade provavelmente está realizando algum tipo de manobra que possui uma distância de deslocamento irrelevante para um cálculo de distância total percorrida durante um longo período de tempo.

Para a obtenção da distância percorrida durante uma corrida, o módulo utilizará o princípio do Algoritmo 1 que calcula a distância percorrida aproximada do mesmo modo da Fórmula 2.3.

Algoritmo 1: Calcula distância percorrida durante uma Corrida

```

inicio ← tempoAtual;
while getVelocidadeVeiculo() > 3 Km/h do
    somaVelocidade += getVelocidadeVeiculo();
    qntAmostrasVelocidade ++;
tempoTotal ← (tempoAtual – inicio);
velocidadeMedia ← (somaVelocidadeCarro / qntAmostrasVelocidade);
distanciaPercorrida ← (velocidadeMedia * tempoTotal);
return distanciaPercorrida

```

Na sessão 3.4.5 sobre a máquina de estados, estão discutidas a interação deste módulo com os demais, assim como o funcionamento da própria máquina de estados, que coordena as invocações de todos os módulos do sistema.

3.4.3 Calculador de Média da Posição da Válvula de Aceleração

Semelhantemente ao módulo de distância percorrida, o Calculador de Média da PVA também é acionado somente no período em que o estado do veículo se encontra em movimento (durante uma corrida), conforme podemos observar na Figura 6. A principal diferença é que não será necessária a utilização de um contador, já que precisamos somente do valor médio da PVA durante a corrida e não depende do tempo. O valor médio pode adquirido do mesmo modo do algoritmo 2:

Algoritmo 2: Calcula média da PVA durante uma Corrida

```

while getVelocidadeVeiculo() > 3 Km/h do
    somaPVA += getPVA();
    qntAmostrasPVA ++;
mediaPVA ← (somaPVA / qntAmostrasPVA);
return mediaPVA

```

Após o termino da corrida, os valores obtidos da média da PVA, distância percorrida, data da corrida, hora de início e hora de término são armazenados no banco de

dados no modelo da tabela *Corrida*, conforme ilustrado pela Figura 5.

3.4.4 Detector de Abastecimento

O módulo de detecção de abastecimento possui o papel no sistema de controlar uma das mudanças de estado do veículo, coletar o valor do nível de combustível atual e iniciar o processo de armazenamento dos registros de consumo de combustível no banco de dados do sistema.

A mudança de estado de Parado para Abastecendo é validada quando o valor do nível de combustível atual for maior que o nível de combustível anterior enquanto o veículo estiver parado. Neste momento, as informações contidas na tabela *auxiliar* são convertidas para a tabela final de *registro*, utilizando o valor de nível atual como nível final (V_f) e o valor de nível inicial contido na tabela *auxiliar* como nível inicial (V_i) para o cálculo de consumo de combustível do mesmo modo da Fórmula 2.2.

Após isso, o sistema aguarda até que o abastecimento seja concluído, estabilizando o nível de combustível em um mesmo valor. O tempo adotado para esse evento foi de 7 segundos, ou seja, o valor do nível de combustível precisa ser o mesmo durante 7 segundos para que seja validado o abastecimento. Este valor foi estimado a partir de valores que deveriam ser longos o suficiente para que o abastecimento não fosse precocemente interrompido, e curto o suficiente para que o veículo não iniciasse uma nova corrida antes da validação do abastecimento realizado, por consequência, o valor de 7 segundos foi o que melhor atendeu a estes requisitos durante os experimentos realizados. O algoritmo 3 demonstra como o abastecimento é detectado e os procedimentos realizados após sua detecção:

Algoritmo 3: Detecta o Abastecimento

```

if getNvAtual() > nvAnterior then
    finalizaRegistroAuxiliar(nvAnterior);
    while estabilidade(nvAtual) == false do
        | aguardaUmSegundo();
    | preparaNovoRegistroAuxiliar(nvAtual, DataAtual);
else
    | nvAnterior ← nvAtual

```

No escopo do comando *While* na linha 4 (quatro) do algoritmo 3 podemos observar a invocação da função *estabilidade()*, responsável pela identificação da estabilidade do nível de combustível, consequentemente indicando o término do abastecimento. Seu funcionamento segue a estratégia conhecida *debounce*, que é adotada quando se possui uma entrada de dados não muito confiável (valores variam constantemente) e é necessária a segurança de que este valor seja correto e estável (GANSSE, 2004), como pode ser

observado no algoritmo 4.

Algoritmo 4: estabilidade(*nvCombustivel*)

```

for  $i = 1$  to 7 do
    if nvCombustivel  $\neq$  valores[ $i$ ] then
        valores[ $i$ ]  $\leftarrow$  nvCombustivel;
        return false
    return true

```

Enquanto o vetor *valores*[] não possui todos os elementos com o mesmo valor de entrada, o algoritmo não classificará a entrada como estável. A dimensão do vetor é igual a 7 pois esta função é invocada a cada segundo no algoritmo 3, logo, assume-se que o valor de nível de combustível será o mesmo durante o período de 7 segundos.

3.4.5 Máquina de Estados

Para melhor entendimento do funcionamento do sistema, foi implementada uma máquina de estados finitos, com o intuito de mapear todo o comportamento básico do sistema nas realizações das iterações dos módulos de forma síncrona. A lógica proposta na máquina de estados tem o objetivo de resguardar o sistema de possíveis equívocos e falhas desta natureza.

A estrutura da máquina de estado está ilustrada na Figura 6, na qual podemos observar o comportamento dos módulos associados aos estados do veículo. A codificação desta máquina de estado foi executada a partir do algoritmo 5:

O algoritmo 5 representa a convergência dos algoritmos 1, 2 e 3, executando todos os módulos passivos de forma totalmente síncrona, tornando-o apto e eficiente na captura das informações de consumo de combustível do veículo.

O corpo do algoritmo conta com 6 (seis) funções abstratas: *aguardaUmSegundo*, *iniciaUmaCorrida*, *salvaCorrida*, *estabilidade*, *finalizaRegistroAuxiliar* e *preparaNovoRegistroAuxiliar*. Mesmo possuindo uma nomeação com linguagem natural, suas funções e responsabilidades estão explicitadas a seguir:

***aguardaUmSegundo()*:** Responsável por uma simples pausa do sistema em um período de 1 (um) segundo;

***iniciaUmaCorrida()*:** Cria uma nova instância da entidade *Corrida*, armazenando o instante de tempo em que a corrida é iniciada;

***salvaCorrida()*:** Armazena a instância da entidade *Corrida* no banco de dados, associando o registro temporário atual do contexto a ela;

Algoritmo 5: Assistente de Consumo de Combustível

```

while conectado do
    aguardaUmSegundo();
    velAtual ← getVelocidadeVeiculoOBD;
    nvCombAtual ← getNvCombustVeiculoOBD;
    pvaAtual ← getPosicaoValvAceleracaoVeiculoOBD;
    switch estado do
        case PARADO do
            if velAtual > 3 Km/h then
                iniciaUmaCorrida();
                estado ← MOVIMENTO;
            if nvCombAtual > nvCombAnterior then
                finalizaRegistroAuxiliar(nvAnterior);
                estado ← ABASTECENDO;
            else
                nvCombAnterior ← nvCombAtual;
        case MOVIMENTO do
            if velAtual > 3 Km then
                somaVelocidade += velAtual;
                somaPVA += pvaAtual;
                qntAmostras ++;
            else
                tempoTotal ← (tempoAtual – inicioCorrida);
                velocidadeMedia ← (somaVelocidadeCarro/qntAmostras);
                distanciaPercorrida ← (velocidadeMedia * tempoTotal);
                mediaPVA ← (somaPVA/qntAmostras);
                salvaCorrida(distanciaPercorrida, mediaPVA);
                estado ← PARADO;
        case ABASTECENDO do
            if estabilidade(nvCombAtual) = true then
                preparaNovoRegistroAuxiliar(nvCombAtual);
                estado ← PARADO;

```

estabilidade(nvCombAtual): Exatamente a função demonstrada no algoritmo 4;

finalizaRegistroAuxiliar(): Conversão da entidade *Auxiliar* para a entidade *Registro*, realizando o cálculo de consumo de combustível do mesmo modo da Fórmula 3.1 e persistindo este registro no banco de dados;

preparaNovoRegistroAuxiliar(): Anula todos as informações contidas anteriormente na entidade *Auxiliar* do contexto atual (exceto o veículo, usuário e tipo de combustível) e excluindo dos os registros das entidade *Corrida* associados a ele. Após isso, a data do instante e o nível de combustível atual do veículo são atualizados,

preparando o sistema para as próximas capturas de dados;

3.4.6 Interface

O último dos módulos que compõe o sistema é a Interface. Este módulo é responsável por permitir a interatividade do usuário com o sistema, almejando uma comunicação de forma mais clara e intuitiva possível.

Buscando uma melhor experiência no uso do sistema, o desenvolvimento da interface do aplicativo empenhou-se em seguir o guia de implementação visual recomendado pelo Google: *Material Design*.

Segundo Lecheta (2016), *Material Design* trata-se da tentativa do Google de padronizar um guia completo sobre como implementar o visual, animação e interação entre os componentes de um *layout*, tornando o visual do aplicativo consistente, simples e intuitivo. A documentação oficial pode ser encontrada no site oficial do guia¹. Nas versões mais recentes do Android, é possível observar diversas API's para o auxílio do desenvolvimento de interfaces ricas em fluidez, como as presentes nos aplicativos nativos Google. Estas API's foram criadas e disponibilizadas para os desenvolvedores, de modo que a compatibilidade com os dispositivos com versões antigas ainda seja preservada.

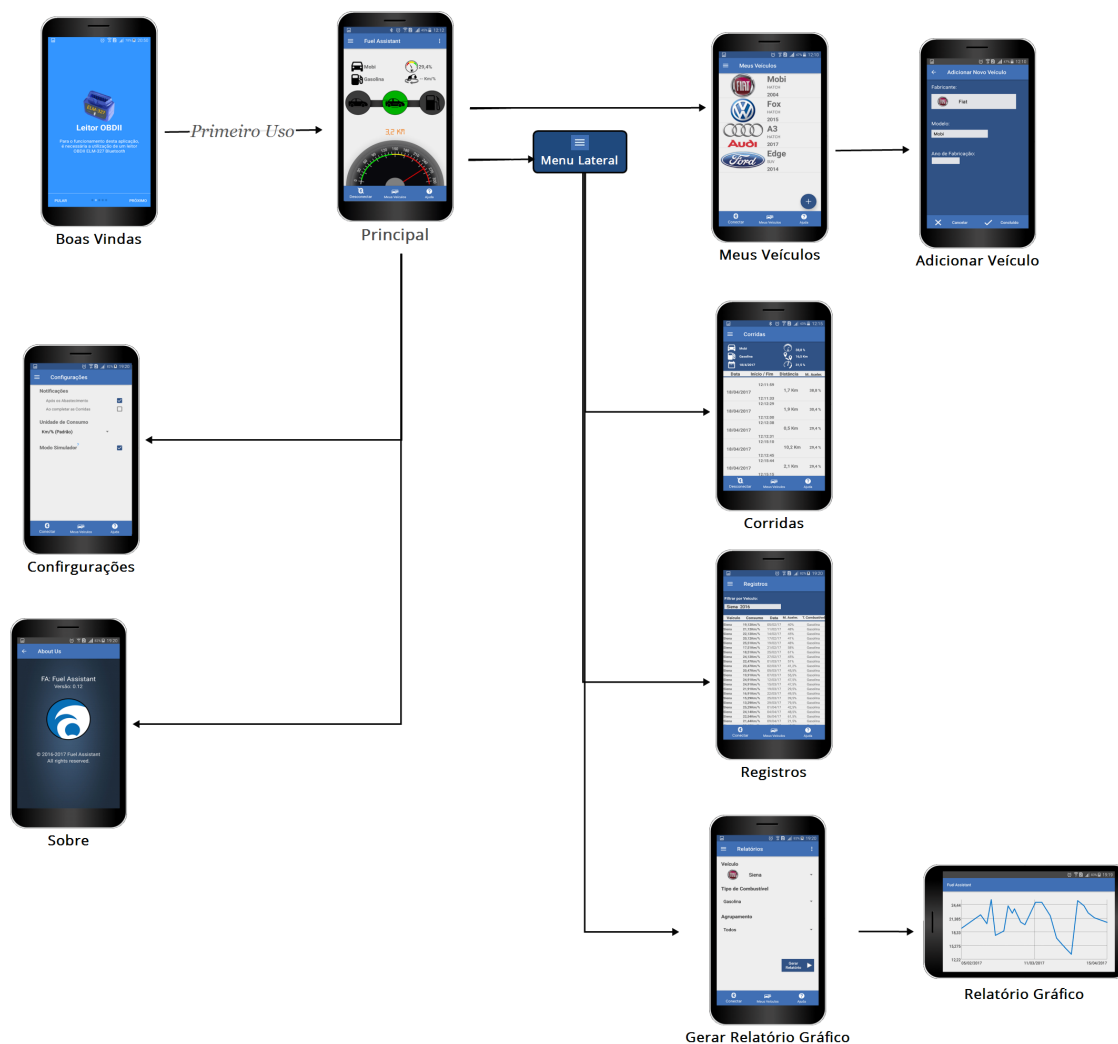
A Figura 8 ilustra o Diagrama de Navegação de Telas do sistema. Este tipo de fluxograma serve para definir as sequências interativas do sistema, apresentando suas principais telas interativas acessíveis.

A interface adota três tipos de menus de acesso às telas: Menu lateral (*Sliding Menu*), Barra Inferior (*Bottom Bar*) e o Menu padrão simples: Pelo Menu lateral é possível acessar as telas *Principal*, *Corridas*, *Registros*, *Relatórios* e *Configurações*. Pela barra inferior é possível acessar o botão que realiza a conexão e desconexão do leitor OBD-II *Bluetooth*, a tela *Meus Veículos* e opção de Ajuda. O menu padrão simples oferece o acesso as telas de *Sobre* e *Configurações*. A seguir estão apresentados os conceitos de cada tela contida na Figura 8:

Principal: Tela inicial do aplicativo, funciona como uma espécie de computador de bordo, contendo um velocímetro que indica a velocidade instantânea do veículo, a distância percorrida durante o registro auxiliar atual, o veículo utilizado, o tipo de combustível utilizado, o nível de combustível instantâneo do veículo e o consumo de combustível obtido até o momento. Além disso, possui um componente que indica o estado atual do veículo (parado, em movimento ou abastecendo).

¹ <https://developer.android.com/design/material/index.html?hl=pt-br>

Figura 8 – Diagrama de Navegação de Telas



Fonte: Autor.

Corridas: Sua função é listar todas as corridas obtidas pelo veículo do contexto atual e seus detalhes, como Data da corrida, horário de início e termino, distância percorrida e *PVA* média. As corridas podem ser gerenciadas nesta tela, como excluí-las ou até transferi-las para outro veículo cadastrado (caso a corrida for registrada erroneamente). Ademais, está tela traz também algumas informações do registro temporário, como data de abastecimento, nível de combustível inicial, distância total percorrida e média de *PVA* total.

Registros: Possui uma lista dos registros de consumo já armazenados no sistema, contendo alguns detalhes como veículo utilizado, o consumo obtido, a data que o registro foi feito, a média da *PVA* e o tipo de combustível. No canto superior da tela há um campo onde se pode filtrar os dados da lista por veículo. Também nesta tela, os

registros podem ser excluídos individualmente.

Meus Veículos: Simplesmente a lista de todos os veículos cadastrados pelo usuário, permitindo a seleção do veículo para o contexto do sistema e exclusão de veículo. Também há um botão para direcionar para a tela de cadastramento de um novo veículo.

Adicionar Veículo: Possui os campos necessários para o cadastro de veículo: Fabricante, Modelo e Ano de Fabricação, além dos botões de cancelamento e confirmação do cadastro.

Gerar Relatório Gráfico: Esta tela permite que seja selecionado o veículo, o tipo de combustível e o agrupamento dos registros cadastrados no sistema para que seja gerado uma relatório de consumo gráfico. O agrupamento possui as opções de agrupamento semanal, mensal, anual ou sem nenhum agrupamento. Após a seleção dos dados, o botão *Gerar Relatório* deve ser utilizado para o direcionamento para o relatório gráfico.

Relatório Gráfico: Resultado da Tela *Gerar Relatório Gráfico*. O gráfico gerado é em forma de linhas, no qual o eixo Y representa o consumo de combustível e o eixo X representa a data de registro.

Configurações: Permite a alteração de alguns componentes do sistema, como a política de notificações, unidade de consumo de combustível, entre outros.

Sobre: Apresenta as informações sobre a versão do sistema e políticas de privacidade.

Boas Vindas: Trata-se de uma sequência slides de instruções para utilização e configuração do sistema. Somente é exibido na primeira utilização do aplicativo.

3.5 Dificuldades Encontradas

Durante o desenvolvimento do trabalho, diversas dificuldades e contratempos foram encontrados, principalmente nas escolhas dos sensores OBD-II, submetendo o trabalho por diversas modificações em relação ao escopo inicial do pré-projeto.

A primeira delas foi a ausência de um sensor OBD-II específico para se obter o valor do odômetro do veículo (equipamento destinado a medir a distância percorrida pelo veículo), já que no escopo inicial, o cálculo do consumo seria realizado simplesmente pelo razão da diferença entre o valor do odômetro inicial e final pela diferença entre o nível de combustível inicial e final, do mesmo modo da equação 3.1:

$$C = \frac{O_f - O_i}{(V_i - V_f)}$$

Consumo de combustível utilizando o valor do odômetro, tal que C representa o Consumo, O_f e O_i os valores do odômetro final e inicial respectivamente e V_i e V_f os valores do nível de combustível inicial e final. (3.1)

Esta seria a maneira mais natural de se calcular o consumo, no entanto durante o desenvolvimento do trabalho, se descobriu que o OBD-II não oferecia suporte para se obter o valor do odômetro. Este sensor já esteve disponível no padrão OBD-II, mas foi retirado em meados de 2012, devido ao grande aumento do número de fraudes de adulteração do odômetro, a fim de reduzir o seu valor para que o veículo seja mais valorizado no mercado, como podemos ver em [Sparks e NHTSA \(2012\)](#).

Outra dificuldade encontrada foi a escassez de veículos que oferecessem suporte para leitura do nível de combustível atual do veículo, já que somente veículos de luxo ou aqueles produzidos nos últimos anos poderiam conter esse tipo de sensor, pois a grande maioria dos veículos utiliza o método mecânico do uso de boias no tanque de combustível que seguem o mesmo princípios das boias de caixa d'água ([COSTA, 2002](#), p. 36), não possuindo um sensor específico para esta finalidade conectado à rede CAN.

4 Resultados

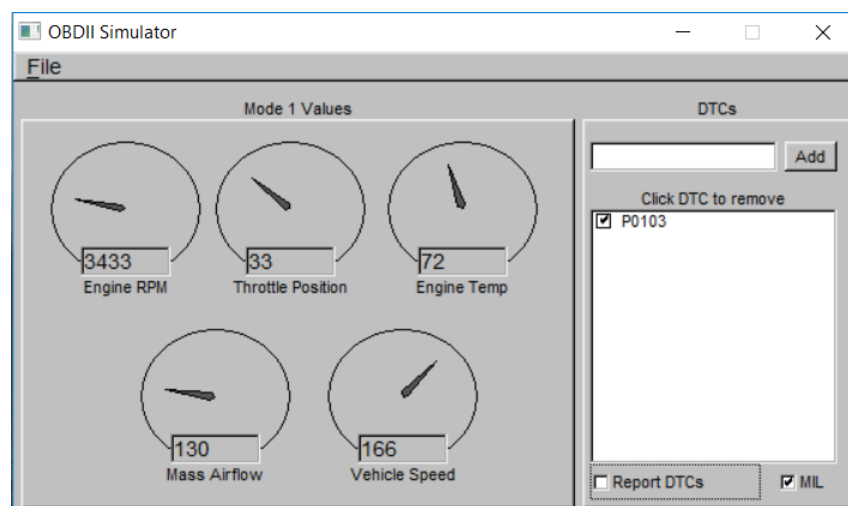
Neste capítulo são apresentados os resultados dos testes realizados no sistema, algumas observações sobre o comportamento do sistema mediante a experimentação e a comparação destes resultados com outras alternativas de obtenção do consumo de combustível.

4.1 Testes

4.1.1 Testes em Laboratório

Durante todo o desenvolvimento do sistema, foi utilizado o simulador OBD-II *Bluetooth* para Windows chamado *OBDSim*¹. Este simulador permite que o usuário possa reproduzir uma conexão *Bluetooth* semelhantemente ao leitor OBD-II ELM 327 através de uma porta COM, possibilitando o envio de pacotes no padrão OBD-II referentes a RPM (Rotações por minuto) do motor, posição da válvula de aceleração (PVA), temperatura do motor, fluxo de massa de ar (MAF) e velocidade do veículo por meio de uma interface gráfica, conforme demonstrado na pela Figura 9

Figura 9 – Interface gráfica do simulador OBDSim



Fonte: Autor.

Como o simulador não possui suporte para a simulação do nível de combustível, requisito primordial para o funcionamento do sistema, durante a realização dos testes

¹ <https://icculus.org/obdgpslogger/obdsim.html>

em laboratório, consideramos na lógica da codificação que o nível de combustível seria representado pelo sensor de temperatura do motor, possibilitando assim verificação do método de cálculo de consumo de combustível.

Estão descritos nesta sessão alguns testes realizados com a versão final do sistema. Os materiais utilizados para a realização destes testes foram:

- Dispositivo Samsung Galaxy Grand Prime SM-G530BT
- Android 5.0.2 *Lollipop* ELM-327
- Notebook Dell Inspiron 15R 7520:
 - Intel Core i5-3230M 2.60GHz
 - 6,00 GB Memória RAM
 - Qualcomm Atheros Wireless Bluetooth 1703
 - Windows 10 Home Single Language
- OBDSim Windows versão de 11/06/2011

O valor temperatura do motor, que está representado o valor de nível de combustível, foi inicialmente alterado para 97 através do OBDSim, simulando que o veículo está com o tanque de combustível a 97% de sua capacidade. Após isso, foram realizadas simulações de corridas, alterando o valor de velocidade do veículo e PVA de forma arbitrária, os valores obtidos pelo sistema para cada corrida realizada, estão dispostos na tabela 2:

Tabela 2 – Resultados das corridas simuladas.

Nv. Combustível (%)	Distância (Km)	Media PVA (%)	Consumo (Km/%)	Duração (min)
97	0,0	0,0	0,0	—:—
95	11,5	23,9	5,75	10:18
92	16,6	22,1	5,53	14:02
90	8,2	29,3	4,1	06:06
81	61,1	26,6	6,78	33:19
75	21,2	27,0	3,53	16:59
59	81,6	33,2	5,1	53:45
<i>Total</i>	200,2	27,0	5,27	

Fonte: Autor.

Após o término da sexta corrida, a distância percorrida total pelo veículo foi de 200,2 Km, utilizando um total de 38% do combustível máximo suportado pelo tanque de combustível do veículo, como ilustrado pela captura de tela da Figura 10. Desta forma, o

Figura 10 – Captura de tela da aplicação após a realização do teste



Data	Início / Fim	Distância	M. Aceler.
19/05/2017	18:34:28 18:24:09	11,5 Km	23,9 %
19/05/2017	18:51:46 18:37:46	16,6 Km	22,1 %
19/05/2017	18:59:02 18:52:58	8,2 Km	29,3 %
19/05/2017	19:34:52 19:01:33	61,1 Km	26,6 %
19/05/2017	19:59:24 19:42:25	21,2 Km	27 %
19/05/2017	20:54:14 20:00:59	81,6 Km	33,2 %

Fonte: Autor.

consumo de combustível atingiu o valor de 5,27 Km/%, correspondendo ao valor esperado pela Fórmula 2.2.

Com o prosseguimento do teste, o nível de combustível foi alterado de 59% para 70% de maneira gradativa, de forma que emulasse a inserção de combustível pela bomba de combustível. Ao iniciar esta ação, a aplicação já demonstra que detectou o abastecimento e que está aguardando até que o nível de combustível se estabilize em um valor durante um determinado momento. Após alguns segundos houve a confirmação do sistema que o abastecimento foi concluído e o registro de consumo realizado foi gravado no banco de dados com sucesso, conforme ilustrado na Figura 11:

4.1.2 Testes de Campo

O parâmetro de comparação do resultado do teste será realizado através da aplicação *Etiquetagem Veicular*, desenvolvido pela Petrobras e que realiza a consulta aos veículos que têm o selo de Eficiência Energética Veicular do CONPET (Programa Nacional da Racionalização do Uso dos Derivados do Petróleo e Gás Natural). Com essa ferramenta podemos determinar o parâmetro de consumo de combustível esperado para o veículo em

Figura 11 – Mensagem de confirmação do sistema após a realização de um abastecimento



Fonte: Autor.

que os testes do sistema deste trabalho serão submetidos. Os materiais utilizados para a realização do teste foram:

- Dispositivo Samsung Galaxy Grand Prime SM-G530BT;
- Android 5.0.2 (*Lollipop*)
- Dongle ELM 327 Mini *Bluetooth*
- Automóvel Jeep Renegade 2016 com motor 1.8 de 139 CV;
- Gasolina comum.

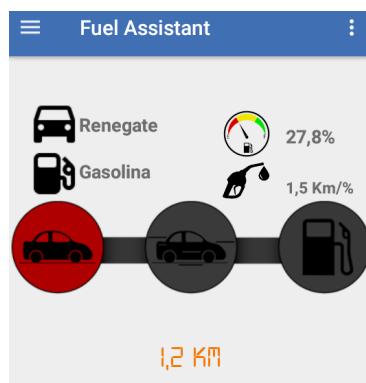
O local do teste foi no município de Formiga - MG, no bairro São Luiz, com o sistema de ar condicionado do veículo desligado. No momento da conexão do sistema ao veículo, verificou-se através do aplicativo que o nível de combustível presente no veículo era de 28,6%. A duração do teste foi de 4 minutos e 34 segundos, no qual o veículo percorreu a distância de 1 Km e 200 metros (1,2 Km) pelo trajeto ilustrado pela Figura 12 com a velocidade média de 16,7 Km/h.

Ao término do teste, o nível de combustível indicado na aplicação era de 27,8%, logo o nível consumido é a diferença entre o nível inicial e atual (28,6 - 27,8) que é igual ao volume de 0,8%.

Portanto, o consumo de combustível apresentado durante o teste, de acordo com a Fórmula 2.2 foi 1,5 Km/% (Quilômetros por nível de combustível), da mesma forma do que foi apresentado pelo sistema, conforme a captura de tela da Figura 13.

Para a comparação do consumo obtido ao consumo esperado para este veículo, é fundamental a conversão da unidade de Km/% para Km/l (Quilômetros por litro). Para isso, é necessário converter o volume de % para litros (l), calculando o produto do nível de combustível em % e da razão entre 100 e a capacidade volumétrica do tanque de combustível do veículo, do mesmo modo da Equação 4.1:

Figura 13 – Captura de tela da aplicação durante o teste realizado.



Fonte: Autor.

Tabela 3 – Valores de referências e resultados obtidos.

Distância (Km)	Combustível Consumido (l)	Consumo (Km/l)	Consumo Esperado (Km/l)	Variação (%)
1,2	0,48	2,5	7,0	450

Fonte: Autor.

ser justificado pela curta duração do teste, que não possibilitou uma amostra variada de valores para a formação do resultado.

Além disso, o trajeto executado trata-se de um via urbana com uma grande quantidade de trechos em aclave (todos na direção sudoeste do trajeto), exigindo uma maior esforço do motor e consequentemente consumindo mais combustível, como podemos observar na Figura 12, em que no momento da transição para um destes trechos, o nível de combustível registrado foi alterado de **28,6%** para **27,6%**. Finalmente, devido a densidade do trânsito no horário do teste, a velocidade média do veículo durante o percurso foi de apenas **16,7 Km/h**, o que proporcionou um baixo desempenho em economia de combustível.

4.2 Comportamento do Sistema no Teste de Campo

Durante a realização do teste de campo, todos os módulos que compõe o sistema se comportaram de maneira esperada, exceto o módulo de detecção de abastecimento que não conseguiu obter o resultado esperado, pois alguns veículos são abastecidos somente enquanto o sistema elétrico está desligado, já que na maioria dos modelos, a chave utilizada para acionar o veículo é necessária para abrir a proteção de entrada do tanque de combustível.

No teste realizado, observou-se que o veículo de teste exigia (para o funcionamento da rede CAN e todos os seus sensores) que a chave estivesse ligada, acionando todo o

sistema elétrico para extração de dados, portanto, o planejamento da máquina de estados da Figura 6 e o algoritmo 3 deverão ser remodelados, para terem a compatibilidade com veículos em que a interrupção da conexão com OBD-II para o abastecimento é necessária.

Para este caso, uma solução viável seria o armazenamento do último valor de nível de combustível coletado do veículo antes de uma desconexão com o sistema. Após a próxima conexão, o valor do nível de combustível anterior seria comparado ao nível de combustível atual. Caso houver divergência neste valores, consistiria em um abastecimento ocorrido durante o período em que o veículo esteve desconectado, do mesmo modo como demonstrado no Algoritmo 6:

Algoritmo 6: Verifica se houve abastecimento durante a desconexão com sistema

```

nvCombAtual ← getNvCombustVeiculoOBD;
if nvCombAtual > nvAnterior then
    |   finalizaRegistroAuxiliar();
    |   preparaNovoRegistroAuxiliar(nvCombAtual, DataAtual);
maquinaDeEstados();
nvAnterior ← nvCombAtual;
finalizarDesconexão();

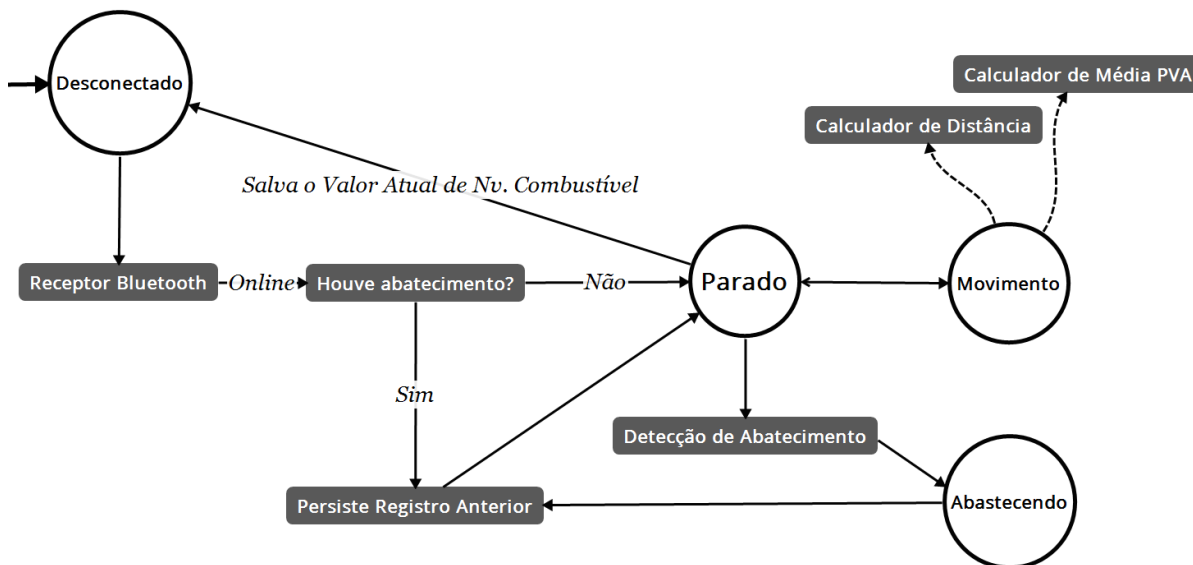
```

O algoritmo 6 deve ser invocado após a instauração da conexão do sistema ao veículo. Sua responsabilidade é garantir a detecção de um abastecimento realizado enquanto o veículo esteve desconectado do sistema, além de invocar o método da máquina de estados e armazenar o último nível de combustível coletado, antes de desconectar do veículo.

Nota-se que o algoritmo 5 da máquina de estados, poderá permanecer inalterado para esta solução, já que sua execução se trata de um laço de repetição que termina quando o sistema for desconectado. Assim, após a sua realização, o último valor de nível de combustível coletado é armazenado, para que seja comparado com o nível coletado durante a próxima conexão do sistema ao veículo.

Após esta modificação, a máquina de estado será definida de acordo com a Figura 14, acrescentando a função de detecção de abastecimento durante o período em que o sistema esteve desconectado do veículo, sem abandonar o método de detecção em tempo real, garantindo a compatibilidade do sistema com os veículos que necessitam desligamento de seu sistema elétrico para a realização dos abastecimentos e os veículos que podem mantê-lo ligado.

Figura 14 – Máquina de Estados após a inclusão da detecção de abastecimentos durante a desconexão do sistema



Fonte: Autor.

4.3 Verificação do Método

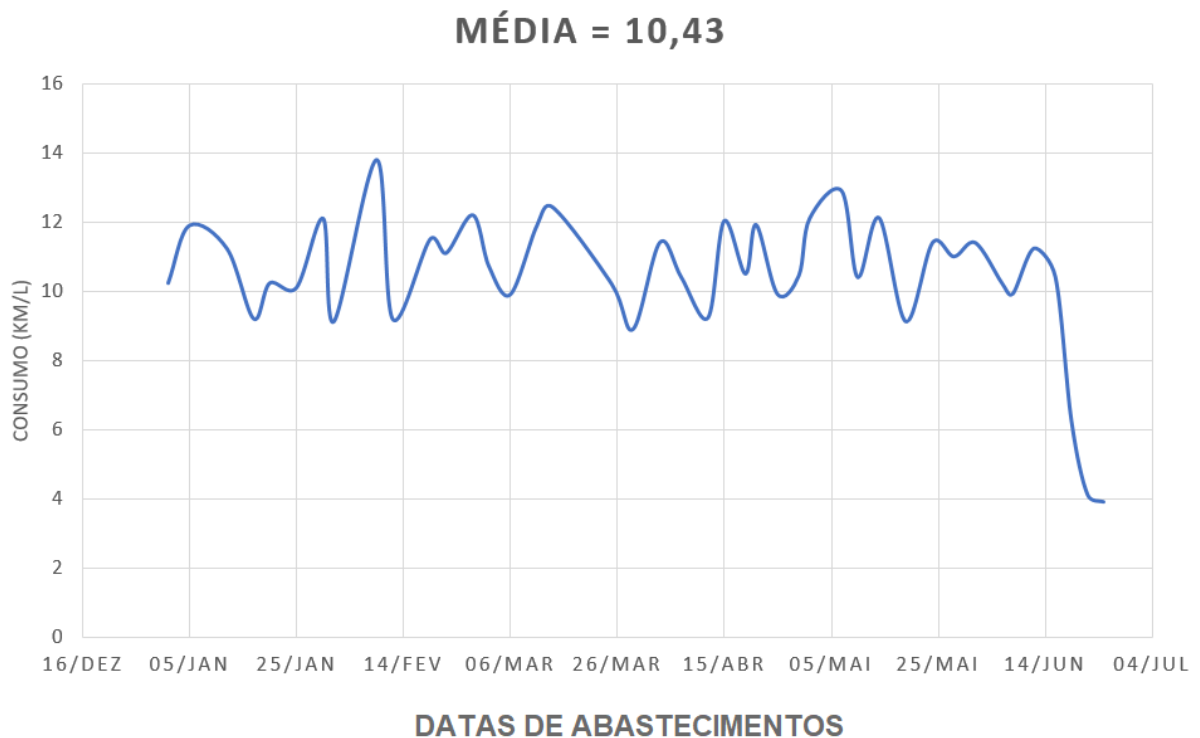
A forma na qual o sistema deste trabalho calcula o consumo de combustível foi apresentada no capítulo 2.1, porém, existem diversas outros métodos para se obter o consumo de combustível de um veículo, como por exemplo, o método apresentado por [Fontes e Figueiredo \(2014\)](#) utilizado em seu projeto: *Econodrive*. Este método consiste em determinar uma aproximação do consumo de combustível instantâneo a partir de uma equação que envolve os valores de massa de fluxo de ar (*Mass Air Flow* ou MAF) e a velocidade instantânea do veículo. A partir do consumo instantâneo, o sistema exibe o consumo instantâneo médio do veículo durante a sua utilização.

Além do fato do método utilizado no *Ecodrive* seja uma aproximação do valor absoluto, a utilização da média do consumo instantâneo pode prejudicar na detecção de irregularidades nos valores de consumo de combustível se a quantidade de elementos na amostra for muito grande.

Em sistemas que apresentem ao usuário somente o valor médio do consumo, a situação em que há diversos valores de amostra e o valor de consumo comece a assumir valores decadentes, a média geral permanecerá assumindo valores distantes do consumo atual.

Porém como o sistema deste trabalho associa o consumo de combustível aos momentos em que o veículo é abastecido, esta detecção se torna eficiente, pois quando o

Figura 15 – Exemplo de consumo de combustível durante o período de 6 meses



Fonte: Autor.

consumo começa a assumir valores inferiores aos obtidos anteriormente pelo veículo, esta divergência ficará destacada pela apresentação das informações de consumo em relação ao tempo, do mesmo modo da Figura 15.

Além disso, uma medida de consumo absoluta, como apresentada pela Fórmula 2.1, mostra-se mais confiável do que medidas providas por aproximações feitas a partir de valores obtidos por fórmulas que envolvem o comportamento mecânico do veículo, pois o funcionamento dos motores podem variar, dependendo de seus fabricantes, tornando o cálculo de consumo imprevisível e inexato para estes modelos.

5 Conclusão

Desde a idealização deste trabalho, tínhamos em mente o potencial que ele poderia alcançar no futuro, desde que sua fundamentação fosse bem modelada e construída. Em virtude disso, optamos por primeiramente estruturar um conceito mais simples e limitado, com ênfase na captura de informações confiáveis e relevantes sobre o consumo de combustível veicular de modo que viabilizasse a oportunidade de sua expansão para modelos mais robustos e inovadores.

No decorrer do projeto, observou-se que o aprendizado da quantidade de tecnologias envolvidas, além dos percalços encontrados durante o desenvolvimento e testes do trabalho, mostrou que a delimitação do mesmo no início foi acertada, pois, demandou tempo e grande esforço para alcançar as metas estipuladas no cronograma.

O teste de campo realizado foi de grande proveito na consolidação do conceito e verificação do modelo, já que foram observados possíveis modificações para a compatibilidade do sistema com os diversos tipos de veículos disponíveis no mercado.

O aumento da popularidade de sistemas que exploram o uso do OBD-II, fomentará o investimento das indústrias automobilísticas na adequação dos veículos às exigências de funcionamento destas ferramentas, por consequência, a tecnologia OBD-II tende a crescer e se aperfeiçoar cada vez mais nos próximos anos.

A potencialidade do projeto se deve a apresentação resultados pertinentes sobre a coleta de informações de consumo de combustível, além de se tratar de um modelo que requer somente um leitor OBD-II e um *smarthphone*, que é acessível ao público geral por um baixo custo de aquisição.

5.1 Trabalhos Futuros

Para a continuidade deste trabalho, seria adequado a realização de mais testes de validação, preferencialmente de longa duração e em trechos de alta velocidade, com o propósito de avaliar e ajustar os métodos de coleta das informações de consumo de combustível.

Existem diversas aplicações que podem ser desenvolvidas a partir deste trabalho. Uma das mais relevantes é a portabilidade do sistema para as plataformas *iOS* e *Windows Phone*, que representam uma grande parte no mercado de sistema móveis.

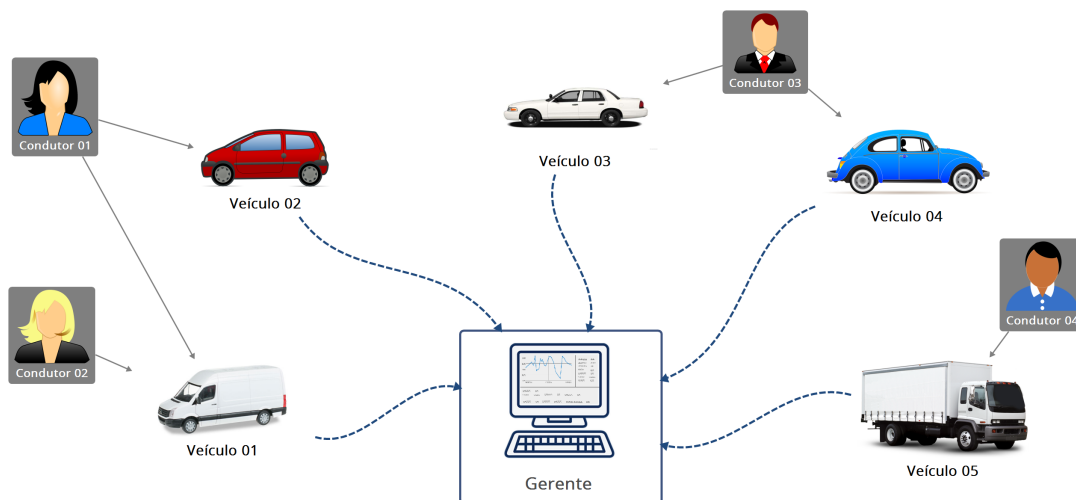
Desde o início do desenvolvimento do trabalho foi notado um potencial de expansão deste sistema, que consistia na centralização dos dados de consumo em um banco de

dados de um sistema computacional, ao invés de serem armazenados no banco de dados do dispositivo Android, permitindo receber as informações de consumo de diversos veículos simultaneamente, e a produção de relatórios envolvendo a comparação de consumo entre veículos, condutores, fabricantes, modelos e etc. Desta forma, foram idealizado dois conceitos de expansão: Corporativo e Global.

5.1.1 Conceito Corporativo

Este conceito se aplica a empresas e corporações que possuem uma frota de veículos, administrada e por um gerente de frota. Cada motorista da corporação utilizaria o mesmo sistema descrito neste trabalho em seus respectivos *smarthphones*, porém com a substituição da persistência dos dados no dispositivo móvel, para a transmissão destes dados via internet para um sistema computacional central utilizado pelo gerente de frota, sem a interação do condutor neste procedimento. A Figura 16 ilustra a estruturação deste conceito.

Figura 16 – Estrutura conceitual do sistema corporativo



Fonte: Autor.

Desta forma, o gerente de frota teria acesso aos dados de consumo de todos os veículos e de todos condutores da corporação, permitindo a obtenção de relatórios úteis na administração de frotas, como a classificação (*ranking*) dos veículos com maior consumo de combustível médio, a classificação dos condutores que possuem o maior consumo médio, a progressão de consumo de combustível individual para cada veículo, modelo de veículo com menor ou maior consumo médio, média da PVA entre os condutores, entre outros.

Com o acesso a estes relatórios, o gerente de frota poderia encarregar-se de tomar as devidas providências de reparação ou manutenção com a finalidade de se obter o consumo

esperado para aqueles veículos que não apresentarem um consumo médio esperado, ou na circunstância, aplicar medidas corretivas sobre os condutores que demonstrarem maus hábitos de condução que influenciaram no consumo de combustível do veículo. Este caso pode ser muito comum, já que os condutores normalmente não se dispõem de cuidados com o consumo de combustível dos veículos de suas corporações, pois não é refletido a eles o custo de aquisição do mesmo. Porém, a relevância desta medida é totalmente notável por parte dos gestores, visto que os gastos com combustível normalmente representam grande parcela do custo total de operação do negócio.

O banco de dados do sistema deste trabalho, descrito no capítulo 3.3, prevê a expansão deste conceito e está totalmente compatível a ela, o que justifica a utilização da tabela *Usuário*, que seria utilizada para identificar o condutor do veículo, e que não é explorada no sistema atual do trabalho.

5.1.2 Conceito Global

O conceito global para esta aplicação emprega o princípio de compartilhamento de informação de consumo, no qual seria oferecido um serviço de armazenamento dos dados de consumo de combustível obtidos por cada usuário em um servidor *web*, ao invés de serem armazenados localmente no banco de dados do dispositivo móvel.

A vantagem desta abordagem é que os próprios usuários do sistema poderiam consultar os valores de consumo de combustível obtidos por outros usuários utilizando configurações semelhantes a ele, como o mesmo fabricante, modelo do veículo, tipo de combustível, cidade ou estado em que o veículo circulou e entre outros, possibilitando diversas possibilidades de mineração destes dados.

Referências

ARRUDA, A. N.; FREITAS, R. E. Renda e despesa familiar no brasil segundo a pesquisa de orçamentos familiares (pof) 2002-2003. *IPEA*, Brasília, 2006. Citado na página 12.

ARRUDA, J. S. J. Desenvolvimento de um método para redução do consumo de combustível no transporte rodoviário de cargas. capacitação, aplicações de torque e telemetria para veículos pesados. *Universidade de Brasília*, Brasília, 2014. Citado na página 12.

AZUGA, D. Danos ambientais causados por veículos leves no brasil. *COPPE/UFRJ*, Rio de Janeiro, 2000. Citado na página 12.

BILLO, E. A. Uma pilha de protocolos bluetooth adaptável à aplicação. *UFSC*, Florianópolis, 2003. Citado na página 23.

CAMPIGLIA, R. B. Estudo da influência da geometria da tubulação a montante da válvula de aceleração no desempenho do sistema de admissão de ar de um motor de combustão interna de ignição por faísca. *USP*, São Paulo, 2007. Citado na página 21.

CERQUEIRA, A. D. et al. Sistema de diagnóstico para veículos que utilizam os protocolos iso 9141 e iso 14230 através de uma plataforma em labview. *FATEC*, Santo André, 2009. Citado na página 21.

CHOW, J. C. et al. Megacities and atmospheric pollution. *Journal of the Air Waste Management Association*, n. 1226-1235, p. 54:10, 2012. Citado na página 12.

CONSIGLIO, T. *Avaliação: Jeep renegade longitude 1.8 flex*. 2015. Revista Auto Esporte. Disponível em: <<http://revistaautoesporte.globo.com/Analises/noticia/2015/04/avaliacao-jeep-renegade-longitude-18-flex.html>>. Acesso em: 19 de mai 2017. Citado na página 49.

COSTA, P. G. A bíblia do carro. 2002. Citado 2 vezes nas páginas 21 e 44.

FONTES, V. G.; FIGUEIREDO, J. A. O. de. Econodrive: Computador de bordo automotivo em plataforma móvel ios. *IFSUL*, Passo Fundo - RS, 2014. Citado 3 vezes nas páginas 13, 22 e 52.

FURRIELA, R. Educação para o consumo sustentável. *USP*, São Paulo, 2000. Citado na página 11.

GANSSE, J. G. A guide to debouncing. *The Ganssle Group*, Baltimore - MD, 2004. Citado na página 38.

GASPAR, A. *Física, volume único*. São Paulo: Ática, 2009. 551 p. (1ª Edição). ISBN 9788508109333. Citado na página 20.

LECHETA, R. *Google Android: Aprenda a criar aplicações para dispositivos móveis com o android sdk*. São Paulo: Novatec, 2016. 1068 p. (5ª Edição). ISBN 9788575224687. Citado 2 vezes nas páginas 24 e 41.

- MACHADO, A. S. L.; OLIVEIRA, B. R. R. O sistema obd (on-board diagnosis). *ISEP*, Porto - Portugal, 2007. Citado na página 22.
- MATTOS, L. B. R. A importância do setor de transportes na emissão de gases do efeito estufa: O caso do município do rio de janeiro. *UFRJ*, Rio de Janeiro, 2001. Citado na página 11.
- MONTEIRO, D. S. Sistema de monitoramento e diagnóstico de veículos com base na plataforma arduino. *Master Christi*, Mossoró - Rio Grande do Norte, 2013. Citado na página 14.
- OLIVEIRA, C. C. Desenvolvimento de banco de dados de Óleos vegetais para aplicação em projeto de produtos. *USP-IME*, São Paulo, 2014. Citado na página 25.
- SANTANA, A. S. J. Sistema de monitoramento eletrônico automotivo. *UniCEUB*, Brasília, 2009. Citado na página 21.
- SILVA, L. P. M. *Consumo de combustível de um automóvel*. 2014. Brasil Escola. Disponível em: <<http://brasilescola.uol.com.br/matematica/calculo-consumo-combustivel-um-automovel.htm>>. Acesso em: 23 de abr 2017. Citado na página 19.
- SOUZA, E. J. de. Manual do condutor de veículos. *ENGETRAN*, Belo Horizonte - MG, 2016. Citado na página 12.
- SPARKS, D.; NHTSA. *Digital Odometer Tampering via OBD II the Controller Area Network*. 2012. SAE International. Disponível em: <<http://www.sae.org/events/gim/presentations/2012/sparks.pdf>>. Acesso em: 23 de abr 2017. Citado na página 44.
- TOMIOKA, J.; SOUZA, M. Obd-iii: Tendências e perspectivas. *UFABC*, São Paulo, 2016. Citado na página 11.

Anexos

ANEXO A – Código Fonte e Aplicativo

O acesso ao código fonte do projeto do sistema deste trabalho pode ser feito utilizando o endereço ¹ ou o *QR Code* da Figura 17, além do arquivo *APK* de instalação em sistemas Android que pode ser baixo através do endereço ².

Figura 17 – QR Codes dos links do arquivo APK de instalação Android e código fonte do sistema.

Instalador APK



<https://goo.gl/NkrYrG>

Código Fonte: Projeto Android Studio



<https://goo.gl/Rdqc3s>

¹ <https://goo.gl/Rdqc3s>

² <https://goo.gl/NkrYrG>