

CENTRO ESTADUAL DE EDUCAÇÃO TECNOLÓGICA PAULA SOUZA
UNIDADE DE PÓS-GRADUAÇÃO, EXTENSÃO E PESQUISA
MESTRADO PROFISSIONAL EM GESTÃO E TECNOLOGIA EM SISTEMAS
PRODUTIVOS

GIOVANI BULGARELLI ZAGO

CAPTURA E DISPONIBILIZAÇÃO EM NUVEM DE LOGS AUTOMOTIVOS: A
CONSTRUÇÃO DE ARTEFATOS DE SOFTWARES PARA COLETA E
GERENCIAMENTO DE ANOMALIAS

São Paulo, maio de 2018

GIOVANI BULGARELLI ZAGO

CAPTURA E DISPONIBILIZAÇÃO EM NUVEM DE LOGS AUTOMOTIVOS: A
CONSTRUÇÃO DE ARTEFATOS DE SOFTWARES PARA COLETA E
GERENCIAMENTO DE ANOMALIAS

Dissertação apresentada como exigência parcial para a obtenção do título de Mestre em Gestão e Tecnologia em Sistemas Produtivos do Centro Estadual de Educação Tecnológica Paula Souza, no Programa de Mestrado Profissional em Gestão e Tecnologia em Sistemas Produtivos, sob a orientação do Prof. Dr. Antonio Cesar Galhardi.

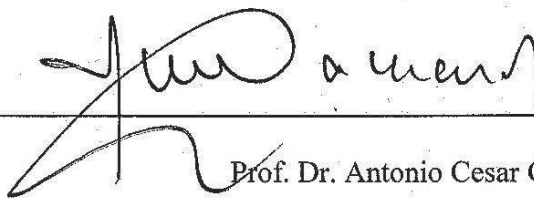
São Paulo, maio de 2018

FICHA ELABORADA PELA BIBLIOTECA NELSON ALVES VIANA
FATEC-SP / CPS

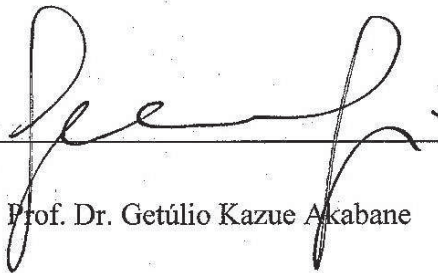
Z18c	<p>Zago, Giovani Bulgarelli</p> <p>Captura e disponibilização em nuvem de logs automotivos: a construção de artefatos de softwares para coleta e gerenciamento de anomalias / Giovani Bulgarelli Zago. – São Paulo : CPS, 2018.</p> <p>73 f. : il., figs.,quads., tabs.</p> <p>Orientador: Prof. Dr. Antonio Cesar Galhardi</p> <p>Dissertação (Mestrado Profissional em Gestão e Tecnologia em Sistemas Produtivos) - Centro Estadual de Educação Tecnológica Paula Souza, 2018.</p> <p>1. Sistemas embarcados automotivos. 2. Computação embarcada. 3. On-Board Diagnostic . 4. Cloud computing. 5. Cadeia de suprimentos automotiva. I. Galhardi, Antonio Cesar. II. Centro Estadual de Educação Tecnológica Paula Souza. III. Título.</p>
------	---

GIOVANI BULGARELLI ZAGO

CAPTURA E DISPONIBILIZAÇÃO EM NUVEM DE LOGS AUTOMOTIVOS: A
CONSTRUÇÃO DE ARTEFATOS DE SOFTWARES PARA COLETA E
GERENCIAMENTO DE ANOMALIAS



Prof. Dr. Antonio Cesar Galhardi



Prof. Dr. Getúlio Kazue Akabane



Prof. Dr. Vivaldo José Breternitz

São Paulo, 24 de maio de 2018

Dedico esta dissertação a Deus.

AGRADECIMENTOS

A Deus, detentor de toda honra e toda glória.

Ao Centro Paula Souza, seu corpo docente, direção e administração por todo auxílio oferecido no decorrer do curso.

Ao meu orientador, Prof. Dr. Antonio Cesar Galhardi, pela oportunidade que me deu, pelo esforço, pela dedicação, pelo suporte, pelo tempo investido, pelas palavras de incentivo e por ter acreditado em meu trabalho.

À minha esposa Gabriela, pela compreensão, companheirismo, dedicação e apoio dado nos momentos de dificuldade.

A todos os meus familiares, que direta ou indiretamente colaboraram para a conclusão deste trabalho.

E aos meus amigos, pela amizade, companheirismo e troca de informações valiosas.

O trabalho só nos cansa se não nos
dedicarmos a ele com alegria.

Rabindranath Tagore

RESUMO

ZAGO, G. B. **Captura e disponibilização em nuvem de logs automotivos**: a construção de artefatos de software para coleta e gerenciamento de anomalias. 73f. Dissertação (Mestrado Profissional em Gestão e Tecnologia em Sistemas Produtivos). Centro Estadual de Educação Tecnológica Paula Souza, São Paulo, 2018.

Ainda que os setores naval, aeronáutico e automobilístico representem setores industriais bastante maduros, a Computação Embarcada tem feito com que a evolução destes setores explore novas formas de mercado. Tradicionalmente, a chamada Computação Embarcada Automotiva foi aplicada para fins relacionados à segurança no setor automotivo, pois, por meio dela, é possível coletar dados relacionados ao funcionamento do veículo em tempo real, podendo assim descobrir falhas no momento em que ocorrem. Com o advento de tecnologias de entretenimento e com a redução de custos de processamento e desenvolvimento de hardwares, a Computação Embarcada Automotiva passou a ser aplicada para inúmeras funcionalidades, tornando-se cada vez mais viável. Dentre os avanços acarretados pela Computação Embarcada Automotiva, situa-se o *On-Board Diagnostic* (OBD), cuja funcionalidade é efetuar o diagnóstico completo e de forma padronizada, do sistema interno veicular, por meio da obtenção dos dados das Unidades de Controle Eletrônico. A tecnologia permite também que por meio de uma interface de comunicação OBD do veículo, os dados obtidos após o diagnóstico sejam transmitidos para o ambiente externo, possibilitando assim que diferentes tecnologias obtenham esses dados e os processem para diferentes finalidades. Dados relacionados à velocidade, combustível, emissão de gases poluentes, temperatura do óleo e códigos de anomalias ocorridos na rede interna do veículo são passíveis de coleta por meio da interface *On-Board Diagnostic*. A exploração desta interface deu-se, inicialmente, pela padronização na forma que os dados são lidos por um dispositivo externo ao veículo. Posteriormente, com o avanço de outras tecnologias, como smartphones e *Cloud Computing*, surgiram novas utilizações para os dados coletados por meio da interface OBD, como: controle de emissão de gases poluentes; previsão de falhas por mal-uso do veículo; prevenção de acidentes por meio de alertas ao condutor e análise de perfil do condutor pela utilização do veículo e de dados de geolocalização. A tendência atual do setor automotivo – principalmente com o advento de tecnologias como *Fog* e *Edge Computing*, responsáveis pela criação de redes virtualizadas entre dispositivos próximos, capazes de efetuar processamentos de baixa latência

– é que os veículos passem a compor uma rede de comunicação, a qual estará em constante troca de dados, de modo a compartilhar suas informações e seu processamento. A presente pesquisa foi desenvolvida considerando as atuais tendências da Computação Embarcada Automotiva, a importância e os avanços na utilização do *On-Board Diagnostic*. Por meio de uma Pesquisa Bibliográfica, as atuais utilizações da tecnologia OBD foram obtidas. A análise dos trabalhos científicos permitiu identificar uma lacuna na grande área da Computação Embarcada Automotiva: explorar a obtenção de códigos de anomalias, coletados de Unidades de Controle Eletrônico (UCE) do veículo por meio da interface OBD, seu armazenamento em um banco de dados na nuvem e a sua disponibilização para eventuais consultas e análises. Como forma de explorar a lacuna encontrada, um conjunto de softwares foi inicialmente sugerido e modificado por meio do método de tentativa e erro. Após algumas alterações em duas diferentes tentativas, descritas no diário de pesquisa, um terceiro modelo de softwares foi desenvolvido. A solução composta por quatro aplicações e um banco de dados foi desenvolvida no sentido de testar a viabilidade de se explorar um grande conjunto de dados de funcionamento dos veículos, armazenados em nuvem; que pudessem ser utilizados com diferentes propósitos, mas, principalmente, que permitissem uma rápida e consistente evolução dos projetos de novos veículos. Após o desenvolvimento do conjunto de softwares, foram efetuados testes e a solução mostrou-se viável para o cumprimento do objetivo proposto, sendo possível por meio dela, a correta coleta de códigos de anomalias veiculares, armazenados nas Unidades de Controle Eletrônico do veículo, o armazenamento destes dados em um banco de dados em nuvem e sua disponibilização por meio de uma Interface de Programação de Aplicações (API). O conjunto de softwares desenvolvidos faz parte do rol de registros do Instituto Nacional da Propriedade Industrial (INPI), da Unidade de Pesquisa do Centro Paula Souza, na construção de sua produção tecnológica.

Palavras-chave: Sistemas Embarcados Automotivos. Computação Embarcada. *On-Board Diagnostic*. *Cloud Computing*. Cadeia de Suprimentos Automotiva.

ABSTRACT

ZAGO, G. B. **Retrieving and delivering automotive logs through the cloud:** building software artifacts for gathering and manage anomalies. 73f. Dissertation (Professional Master in Management and Technology in Productive Systems). State Center of Technological Education Paula Souza, São Paulo, 2018.

Although the naval, aeronautical and automotive sectors represent very mature industrial sectors, Embedded Computing has made the evolution of these sectors explore new forms of market. Traditionally, so-called Automotive Embedded Computing has been applied for safety-related purposes in the automotive sector, because through it, it is possible to collect data related to vehicle operation in real time, thus being able to discover faults as they occur. With the advent of entertainment technologies and the reduction of hardware processing and development costs, Automated Embedded Computing has been applied to many functionalities, making it increasingly viable. Among the advances achieved by Automotive Embedded Computing, is the On-Board Diagnostic (OBD), whose functionality is to perform a complete and standardized diagnosis of the internal vehicular system, by obtaining data from the Electronic Control Units. The technology also allows that through an OBD communication interface of the vehicle, the data obtained after the diagnosis is transmitted to the external environment, thus allowing different technologies to obtain this data and process it for different purposes. Data related to speed, fuel, emission of pollutant gases, oil temperature and fault codes occurring in the vehicle's internal network can be collected through the On-Board Diagnostic interface. The exploration of this interface was initially due to the standardization in the form that the data is read by a device external to the vehicle. Later, with the advancement of other technologies, such as smartphones and Cloud Computing, new uses for the data collected through the OBD interface appeared, such as: emission control of polluting gases; prediction of failures due to misuse of the vehicle; accident prevention through driver alerts and analysis of the driver's profile for the use of the vehicle and geolocation data. The current trend in the automotive sector - especially with the advent of technologies such as Fog and Edge Computing, responsible for creating virtualized networks between nearby devices capable of performing low latency processes - is for vehicles to form a communication network, which will be in constant exchange of data, in order to share its information and its processing. The present research was developed considering the current trends of Automotive Embedded

Computing, the importance and the advances in the use of On-Board Diagnostic. Through a Bibliographic Survey, the current uses of OBD technology were obtained. The analysis of the scientific work allowed to identify a gap in the large area of Automotive Embedded Computing: to explore the obtaining of codes of anomalies, collected from Electronic Control Units (ECU) of the vehicle through the OBD interface, its storage in a database in the available for consultation and analysis. As a way to explore the gap found, a set of software was initially suggested and modified through trial and error. After some changes in two different trials, described in the research journal, a third software model was developed. The solution comprised of four applications and a database was developed to test the feasibility of exploring a large set of vehicle data stored in the cloud; that could be used for different purposes, but mainly to allow a rapid and consistent evolution of new vehicle designs. After the development of the software package, tests were carried out and the solution proved to be feasible for the accomplishment of the proposed objective, being possible through it, the correct collection of vehicular anomaly codes, stored in the Electronic Control Units of the vehicle, storage of this data in a cloud database and its availability through an Application Programming Interface (API). The set of software developed is part of the register of the National Institute of Industrial Property (INPI), of the Research Unit of the Paula Souza Center, in the construction of its technological production.

Keywords: Automotive Embedded Systems. Embedded Computing. *On-Board Diagnostic*. *Cloud Computing*. Automotive Supply Chain.

LISTA DE QUADROS

Quadro 1	Resumo de Publicações Referentes à Utilização Atual do OBD.....	37
Quadro 2	Caracteres para Localização do Código de Falha.....	58
Quadro 3	Análises Efetuadas nos Testes da Solução.....	60

LISTA DE TABELAS

Tabela 1	Código de Anomalia Armazenado no <i>CarLoggerDB</i>	63
----------	---	----

LISTA DE FIGURAS

Figura 1	Estrutura da Pesquisa.....	25
Figura 2	Foto de uma Saída OBDII.....	30
Figura 3	Exemplo de ELM327.....	30
Figura 4	Lâmpadas de Indicação de Defeito de Painéis dos Veículos.....	33
Figura 5	Modelo 4S.....	39
Figura 6	Publicações Relacionadas a ‘On-Board Diagnostic’ por Período.....	48
Figura 7	Modelo Proposto.....	51
Figura 8	Tela de Login.....	52
Figura 9	Menu do Sistema.....	53
Figura 10	Tela de Cadastro.....	53
Figura 11	Tela de Análise.....	54
Figura 12	Detalhes do Retorno com Protocolo CAN.....	56
Figura 13	Detalhes do Retorno com Protocolo KWP.....	57
Figura 14	Resultados Obtidos e Salvos no Banco de Dados.....	60
Figura 15	Resultados Obtidos Por Meio do <i>CarLoggerAPI</i>	61
Figura 16	Resultados da Análise.....	62
Figura 17	E-mail Recebido.....	62
Figura 18	Log Obtido pelo <i>CarLoggerAPI</i>	63

LISTA DE ABREVIATURAS E SIGLAS

4S	<i>Sale, Service, Spare Parts and Survey</i>
ACM	<i>Association for Computing Machinery</i>
AGC	<i>Apollo Guidance Computer</i>
ANFAVEA	Associação Nacional dos Fabricantes de Veículos Automotores
API	<i>Application Programming Interface</i>
BMW	<i>Bayerische Motoren Werke</i>
BUS	Barramentos
CAN	<i>Controller Area Network</i>
CE	Comissão Europeia
CETESB	Companhia Ambiental do Estado de São Paulo
CI	Circuito Integrado
CO	Monóxido de Carbono
CONAMA	Conselho Nacional do Meio Ambiente
CPU	<i>Central Processing Unit</i>
CSV	<i>Comma-Separated Values</i>
C#	<i>C Sharp</i>
DTC	<i>Data Trouble Code</i>
ECU	<i>Electronic Control Unit</i>
EGR	<i>Exhaust Gas Recirculation</i>
EOBD	<i>European On-Board Diagnostic</i>
GPS	<i>Global Positioning System</i>
GST	<i>Generic Scan Tool</i>
HC	Hidrocarboneto
IaaS	<i>Infrastructure as a Service</i>
IBAMA	Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis
IBM	<i>International Business Machines</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IOS	<i>Iphone Operating System</i>
ISO	<i>International Organization for Standardization</i>
KML	<i>Keyhole Markup Language</i>
KWP	<i>Keyword Protocol</i>
LIM	Lâmpada Indicadora de Mau Funcionamento

MIL	<i>Malfunction Indication Lamp</i>
NH3	Amônia
NO	Óxido Nítrico
NO2	Dióxido de Nitrogênio
OBD	<i>On-Board Diagnostic</i>
OBM	<i>On-Board Measurement</i>
PaaS	<i>Platform as a Service</i>
PID	<i>Parameter ID</i>
PROCONVE	Programa de Controle de Poluição do Ar por Veículos Automotores
PWM	<i>Pulse-Width Modulation</i>
SaaS	<i>Software as a Service</i>
SAE	<i>Society of Automotive Engineers</i>
SEBRAE	Serviço Brasileiro de Apoio às Micro e Pequenas Empresas
SQL	<i>Structured Query Language</i>
SRC	<i>Selective Catalytic Reduction</i>
TIC	Tecnologia de Informação e Comunicação
UCE	Unidade de Controle Eletrônico
UE	União Europeia
URL	<i>Uniform Resource Locator</i>
USB	<i>Universal Serial Bus</i>
VANET	<i>Veicular Ad Hoc Networks</i>
VPW	<i>Variable Pulse Width</i>
WCDMA	<i>Wide-Band Code-Division Multiple Access</i>

SUMÁRIO

INTRODUÇÃO.....	17
Justificativa.....	26
Objetivos.....	26
Problema de Pesquisa.....	26
 1 FUNDAMENTAÇÃO TEÓRICA.....	 27
 2 METODOLOGIA.....	 47
 3 RESULTADOS E DISCUSSÕES.....	 48
3.1 Pesquisa Bibliográfica.....	48
3.2 Diário de Pesquisa.....	50
3.3 Teste da Solução.....	59
 CONSIDERAÇÕES FINAIS.....	 64
 REFERÊNCIAS.....	 66
 ANEXOS.....	 71

INTRODUÇÃO

Embora as montadoras, bem como toda a Cadeia de Suprimentos Automobilística representem um setor industrial bastante maduro, recentemente ele tem passado por um período turbulento, no que se refere aos novos projetos de desenvolvimento de produtos, principalmente com a inclusão de itens da computação embarcada automotiva. As principais características dos sistemas embarcados automotivos são: atuam em tempo real e atuam sob várias restrições relacionadas a espaço, peso, consumo de energia e às despesas ocasionadas pela aplicação específica.

Neste sentido, os veículos automotivos gozaram de muitos aspectos dessa revolução, com inovações que vão desde o controle eletrônico de estabilidade, os sistemas de controle de tráfego urbano até os sistemas de coleta eletrônica de portagens na rede rodoviária. As recentes tendências em Tecnologias de Informação e Comunicação (TIC) apontam para uma crescente lista de novas aplicações viáveis.

Muitas tecnologias atuais são praticamente dependentes de sistemas embarcados para seu funcionamento. No setor automotivo, a necessidade de sistemas embarcados ocorre, por exemplo, em alguns aprimoramentos relacionados às funções de segurança e conforto, introduzidas a partir de 2010, como: partida sem chave; sistemas de controle inteligente de suspensão e novos sistemas de assistência ao condutor (WATZENIG; ARMENGAUD, 2011).

Uma das principais aplicações envolve a comunicação entre um grande número de veículos em uma grande escala geográfica, especialmente as redes veiculares, os *Vehicular Ad-Hoc Networks* (VANETS), e o compartilhamento de informações dos diagnósticos e medições a bordo, os *On Board Diagnostic e On Board Measurement* (OBD)/(OBM).

Tradicionalmente, a computação embarcada automotiva foi aplicada principalmente para fins relacionados à dispositivos de segurança, posteriormente é que surgiram aplicações orientadas para o entretenimento. No entanto, à medida que o processamento e as comunicações de bordo tiveram redução de custos e se tornaram menos invasivos, o aumento de escala em aplicações tornou-se viável.

Os veículos atuais possuem um grande número de sensores para o monitoramento de aspectos do funcionamento do motor, como temperatura e consumo de combustível. Alguns sensores efetuam leituras relacionadas às outras unidades, como: sensores de aproximação, que detectam a distância do objeto mais próximo e sensores que detectam o fechamento das portas. Os sensores podem variar a frequência da amostragem e a quantidade de dados que eles captam.

Por exemplo, um sensor de temperatura pode capturar um único byte e ser amostrado em determinado período de segundos, enquanto uma câmera de vídeo pode receber vários megabytes por segundo (GALHARDI, 2014).

Com o passar do tempo os sistemas embarcados evoluíram. Na medida em que os processadores utilizados em sistemas embarcados se tornaram mais potentes e sofisticados, tornou-se viável a utilização de sistemas operacionais para auxiliar o desenvolvimento de softwares embarcados (GALLASSI; MARTINS, 2014).

Alguns sistemas instalados pelo fabricante do veículo usam uma Unidade Central de Processamento (*Central Processing Unit* - CPUs) de uso geral, que operam sistemas operacionais próprios, como por exemplo, o sistema iDrive da *Bayerische Motoren Werke* (BMW), uma interface de software para permitir que o condutor interaja com o veículo; e se utiliza do sistema operacional VxWorks em tempo real (GALHARDI, 2014).

Ao invés desses sistemas serem instalados em caixas pretas separadas, cada uma com suas próprias CPUs, é concebível que, no futuro, os fabricantes de veículos possam construir veículos equipados com CPUs de uso geral, que sejam compartilhados por todas as suas aplicações (GALHARDI, 2014).

Os fabricantes estão preocupados com o ganho de peso dos veículos, a cada inclusão de novas aplicações. No passado, a fiação era aplicada aos veículos para cada novo sensor adicionado. No entanto, à medida que os veículos ganharam um número crescente de sensores, o peso aumentou consideravelmente, ocupou espaço significativo e dificultou a aderência aos padrões de confiabilidade. Assim tiveram origem as CPUs de uso geral (LEEN; HEFFERNAN, 2002).

Isso também levou à adoção de uma abordagem baseada em barramentos (BUS), com fiação compartilhada entre sensores. Em meados da década de 1980, a rede de área do controlador, *Controller Area Network* (CAN) foi desenvolvida, e é ainda amplamente utilizada em alguns veículos modernos (LEEN; HEFFERMAN, 2002).

Segundo Palocz-Andresen (2012) a evolução dos dispositivos eletrônicos veiculares e o grande número de dados sendo obtidos pelos sensores, aliados aos novos protocolos de comunicação desenvolvidos, tornaram-se cada vez mais completos o complexo diagnóstico veicular, mais especificamente no que se refere à padronização das informações. O reconhecimento da necessidade de monitoramento das informações dos veículos, para se atender às Legislações Ambientais, foi responsável pelo desenvolvimento, no fim dos anos 1980, da tecnologia de diagnóstico de bordo, *On-Board Diagnostic* (OBD) e de sua interface de comunicação.

Assim, surgiu o sistema integrado de diagnóstico de bordo: On-Board Diagnostic (OBD) que monitora as operações em execução em determinados intervalos de tempo. Combina sensores, atuadores, fios, baterias, plugues e similares, no veículo. Pode identificar códigos de anomalias que são armazenados na memória e permite a leitura dos dados do sistema interno do veículo, por meio de uma interface padrão de comunicação. A introdução do OBD tornou-se possível devido ao rápido desenvolvimento de microcontroladores e técnicas de engenharia de microsistemas nos últimos anos. Esta tecnologia, até os dias atuais, possui como objetivo padronizar a forma de comunicação com o veículo, mas, inicialmente, só apresentava as seguintes funções básicas:

- Monitoramento dos componentes de emissão conectados ao controlador (predominantemente diagnóstico de condução elétrica).
- Saída de códigos de falhas, por exemplo, o código intermitente do *Malfunction Indication Lamp* (MIL), luz de indicação de mau funcionamento no painel do veículo.

Em busca do aperfeiçoamento da tecnologia OBD, nos anos de 1990 foi desenvolvida uma nova versão, chamada OBDII. A introdução do OBDII seguiu com exigências de grande alcance para:

- Monitoramento de todos os sistemas relevantes para gases de escape e componentes que referem mau funcionamento ou de falha, de importantes sistemas e componentes de redução de emissões (por exemplo, catalisador, sensor lambda).
- Armazenamento de falhas e das condições de operação associadas.
- Leitura da memória de falhas, por meio de um dispositivo de saída padronizado (ferramenta de varredura genérica, a *Generic Scan Tool* - GST).

A partir de 2004, o OBD II tem sido constantemente atualizado. Além de requisitos funcionais intensificados e adicionados, passaram também a estimar a frequência com que os diagnósticos devem ser realizados.

Na Europa, o OBDII também foi uma tecnologia adequada para examinar a operação por meio de sinais de anomalias e reparo simplificado (TAPPE, 1997). A variante europeia deste sistema de diagnóstico é denominada *European On-Board Diagnostic* (EOBD), que também permite combinar diagnósticos com medidas.

Segundo Husni et al. (2016), a tecnologia *On-Board Diagnostic* não apenas possui a capacidade de monitoramento do sistema interno veicular, mas, também, permite que se efetue

a captura dos dados do sistema interno, por meio de uma interface de comunicação localizada próximo ao motorista.

No Brasil, o Conselho Nacional do Meio Ambiente (CONAMA), em sua resolução de número 315/2002, Art. 10º anunciou a obrigatoriedade do porte do dispositivo OBD e pediu para que Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis (IBAMA) propusesse as especificações e datas de implementação, com o foco de reduzir a emissão de gases poluentes.

Após dois anos, o CONAMA 354/2004, em seu artigo primeiro teve como objetivo:

Estabelecer para veículos leves de passageiros e leves comerciais, nacionais e importados, destinados ao mercado brasileiro, equipados com motores do ciclo Otto, a utilização de sistema de diagnose de bordo (OBD) introduzidos em duas etapas consecutivas e complementares denominadas OBDBr-1 e OBDBr-2, em atendimento ao art. 10 da Resolução nº 315, de 29 de outubro de 2002, do Conselho Nacional do Meio Ambiente-CONAMA (BRASIL, 2004).

A etapa denominada OBDBr-1 exigia que: a partir de 1 de janeiro de 2007, 40% do total da produção anual de veículos estivessem adequados às exigências mínimas da etapa; a partir de 1 de janeiro de 2008, 70% dos veículos deveriam estar adequados às exigências e, a partir de 1 de janeiro de 2009, 100% deveriam estar adequados às seguintes características de detecção de falhas para a avaliação do funcionamento dos sistemas de ignição e de injeção de combustível:

- a) sensor pressão absoluta ou fluxo de ar;
- b) sensor posição da borboleta;
- c) sensor de temperatura de arrefecimento;
- d) sensor temperatura de ar;
- e) sensor oxigênio (somente o sensor pré-catalisador);
- f) sensor de velocidade do veículo;
- g) sensor de posição do eixo comando de válvulas;
- h) sensor de posição do virabrequim;
- i) sistemas de recirculação dos gases de escape, *Exhaust Gas Recirculation* (EGR);
- j) sensor para detecção de detonação;
- l) válvulas injetoras;
- m) sistema de ignição;
- n) módulo controle eletrônico do motor;
- o) lâmpada indicadora de mau funcionamento (LIM); e
- p) outros componentes que o fabricante julgue relevantes para a correta avaliação do funcionamento do veículo e controle de emissões de poluentes (BRASIL, 2004).

Foram solicitadas para a adequação ao plano que: dia 1 de janeiro de 2010, 60% dos veículos nacionais ou importados para o mercado interno estivessem adaptados às especificações e, a partir de 1 de janeiro de 2011, a totalidade dos veículos. Em ambas as resoluções da CONAMA, é apresentado que o início, a adaptação e a integração ao projeto de

implementação de diagnóstico de bordo faz parte do Programa de Controle de Poluição do Ar por Veículos Automotores, visando à redução da emissão de gases poluentes por veículos automotores, por meio da análise do correto funcionamento dos componentes responsáveis em veículos para que estejam sempre funcionando adequadamente (BRASIL, 2004).

As tecnologias OBD/OBM acopladas representam um sistema de informação acessível e confiável para medir a produção de poluentes durante a operação. A combinação de ambos os sistemas constitui um instrumento promissor para as crescentes demandas de adesão aos valores limitantes (FRIEDRICH et al., 1998).

As diretrizes da União Europeia (UE) que descrevem a tecnologia combinada OBD/OBM em veículos a motor existem desde 1998. Elas fornecem meio para uma possível redução das emissões dos veículos motorizados. Uma vantagem especial desta tecnologia é a sua capacidade de observar diretamente as emissões dos veículos a motor, enquanto o mesmo esteja em operação.

Para a legislação de controle de poluentes, o OBM possui um objetivo estratégico; no entanto, os avanços futuros em tecnologia e organização do tráfego podem aumentar significativamente suas aplicações como, por exemplo, a maior precisão na detecção da degradação causada pelo envelhecimento das partes. A deterioração da eficiência, em termos de oxidação e redução de poluentes, é continuamente detectada no conversor catalítico, e isto possibilita monitorar, de forma precisa e direta, o filtro regenerativo e as tecnologias para controle da redução de nitrogênio, como a redução catalítica seletiva (*Selective Catalytic Reduction* - SRC) ou a reciclagem de gases de escape.

Uma declaração da Comissão Europeia (CE) referente à legislação sobre os gases de escape salientou que, no que se refere ao monitoramento das emissões de veículos automotores, todos os elementos do sistema apresentam dependência entre si. No entanto, a tecnologia OBD só detecta a qualidade dos componentes individuais dos gases de escape. A consideração adicional de deteriorações múltiplas, por exemplo, pelo envelhecimento paralelo de vários componentes não existia, até o surgimento do OBM (WEAVER; PETTY, 2004)

Enquanto no OBD, um código de anomalia de desvios estatisticamente apurados é armazenado e um sinal de alerta é emitido, o que indica que em breve o veículo deva ser reparado; em contraste no OBM, observam-se emissões diretas e a deterioração dos gases de escape, talvez causados por anomalias inesperadas ou combinações de anomalias detectadas além da deterioração dos elementos relacionados às emissões.

A Directiva 98/69/CE da União Europeia (UE, 1998) contém um anúncio importante e relevante para a época, afirmando que a possibilidade de controle não é apenas possível para

componentes individuais, utilizando uma técnica de medição apropriada como suplemento ao OBD puro. No entanto, OBM não deve ser considerado como um substituto para OBD, mas sim, um elemento complementar.

Assim como a tecnologia OBM teve sua utilização acoplada ao OBD, de modo a aperfeiçoar as análises e cumprir com os requisitos impostos para o sistema de diagnóstico, outras tecnologias passaram a complementar as análises dos dados do sistema interno veicular. Com a possibilidade de obter os dados de diagnóstico de bordo, por meio da interface de comunicação OBD, a utilização de dispositivos móveis e os serviços de geolocalização e computação em nuvem, passaram a integrar diferentes soluções, com o objetivo de efetuar análises para fins específicos.

Cottingham e Davies (2007) apresentam uma variedade de aplicações, algumas das quais já estão disponíveis no mercado, cujas análises dos dados obtidos por meio da interface OBD são utilizadas para diferentes fins:

- Serviços baseados em localização podem incluir dados meteorológicos ou de tráfegos locais, ou anúncios de empresas localizadas na área.

- Informações de congestionamento. Uma unidade de navegação no veículo pode receber a transmissão das condições de tráfego atuais, determinadas, por exemplo, agregando dados coletados dos diversos sensores da via. Além disso, a unidade de navegação pode consultar proativamente um serviço de informações de congestionamento remoto sobre as estradas, e incorporar em uma rota. Esse serviço pode coletar dados de veículos e depois de processados, fornece estimativas mais precisas dos tempos de viagem (DAY; WU; POULLTON, 2006).

- Clima. Da mesma forma, os veículos podem incorporar as observações e previsões climáticas para a região local. Além disso, a rede pode coletar dados de veículos, que contêm sensores meteorológicos e redistribuir os dados agregados aos demais.

- Detecção de perigo na estrada. Os riscos potenciais nas estradas podem ser detectados se os dados dos sistemas de frenagem de vários veículos forem reunidos, e quando um número substancial de veículos frear bruscamente em um local determinado, a ação pode ser determinada como um perigo potencial (GRUTESER; GRUNWALD, 2003).

- Geração de mapas. Da mesma forma, os históricos de localização de um grande número de veículos podem ser combinados e agregados para fornecer atualizações de mapas rodoviários digitais em tempo real. Por exemplo, novas estradas ou fechamentos de estradas

podem ser detectados e a informação entregue de volta aos veículos (DAVIES; BERESFORD; HOPPER, 2006).

Algumas das aplicações descritas anteriormente, por Cottingham e Davies (2007), envolvem a coordenação de computação para processar dados de sensores entre um grande número de veículos. Isso pode tornar o design e a implementação desses aplicativos desafiadores. Especificamente, existem quatro desafios enfrentados pelos projetistas de aplicativos:

1. Os veículos precisam de um meio de comunicação sem fio com outros veículos e com a Internet.

2. A mobilidade dos veículos implica em que as características dos links de comunicação irão variar ao longo do tempo. As aplicações devem ser capazes de se adaptarem a tais mudanças na conectividade.

3. As aplicações têm como meta se relacionar com toda a rede e não com veículos individuais específicos. Isso significa que o conjunto de veículos participantes não será necessariamente conhecido antes da execução do aplicativo. As aplicações devem, portanto, ser capazes de se adaptar a qualquer recurso computacional disponível para executá-lo em tempo real.

4. Uma rede que inclua computadores de bordo de veículos será sempre heterogênea. Os computadores participantes serão diferentes em termos de seus níveis de recursos computacionais, de armazenamento, energia e comunicação. As aplicações devem tornar o uso desses recursos o mais eficiente possível.

A implementação dessas aplicações pode ser estimulada pela existência de plataformas de computação de uso geral em veículos. Os fabricantes de veículos, para evitar a redundância causada por aplicativos ou dispositivos isolados, podem compartilhar software ou hardware comum.

Destas aplicações, uma das mais promissoras e também mais desafiadora envolve a coleta, o processamento e a distribuição de dados de sensores originados em veículos. Nessas aplicações, os veículos se tornam plataformas e nós de sensores em uma rede de sensores sem fio de grande escala.

Softwares de dispositivos móveis para uso doméstico também são muito comuns quando se trata da interface OBD. Muitas aplicações são desenvolvidas com o objetivo de efetuar a análise dos dados do sistema interno veicular e manter o condutor ou o proprietário, informado

em relação às condições gerais de seu veículo. Existem aplicações com as mais variadas funções, como: análise dos códigos de falha; localização de oficinas mecânicas; transformação do smartphone em um computador de bordo; análise de perfil do motorista; gerenciamento de frotas e análise das condições gerais de tráfego com base nos dados de utilização do veículo e geolocalização.

No Brasil, em 2014, foi disponibilizado um aplicativo denominado Carrorama para as plataformas *Iphone Operating System* (IOS) e Android, que a partir das informações obtidas pelo OBD, transforma o celular em um computador de bordo. O dispositivo permite obter, em tempo real, um diagnóstico do veículo, com informações sobre rotação do motor, velocidade, temperatura da água, voltagem da bateria, nível do tanque de combustível, percentual de etanol no tanque de combustível, número do chassi e posição do acelerador. Essas informações nem sempre estão disponíveis nos painéis dos carros populares, mas, estão disponíveis na primeira tela do aplicativo. O acesso aos dados varia de acordo com as informações disponibilizadas por cada montadora (PORTAL G1, 2014). Desenvolvido pela empresa Going2, o Carrorama privilegia as características dos veículos nacionais, como o abastecimento com diversos tipos de combustível.

No final do ano de 2017, o criador do aplicativo Waze lançou uma nova aplicação voltada para o diagnóstico do veículo, chamado *Engie App*. Segundo a marca, o aplicativo permite que o proprietário do veículo conecte o aplicativo de seu smartphone com um dispositivo leitor – vendido pela empresa ou de marca indicado para o uso – conectado ao OBDII do carro, para que com este, sejam feitas análises do veículo. A análise das informações permite não apenas que o usuário da aplicação tenha o conhecimento das falhas ocorridas, para que assim busque pela correção, mas, igualmente, impede que o proprietário seja enganado por mecânicos mal-intencionados, que informem falsas correções, para cobrar valores mais altos, ou que informem a correção do problema, quando na verdade ele ainda está presente e pode ser diagnosticado por meio da aplicação (PORTAL EXAME, 2018).

Além do recurso de identificação de falhas, o aplicativo possui outras funções: localização de oficinas mecânicas nas proximidades, por meio da utilização do *Global Positioning System* (GPS) do smartphone; quais são as oficinas dessa região que tem destaque e, após a identificação de uma falha no veículo, é possível obter, em tempo real, orçamentos em oficinas próximas com o valor exato para a correção do problema encontrado (PORTAL ENGIE, 2018).

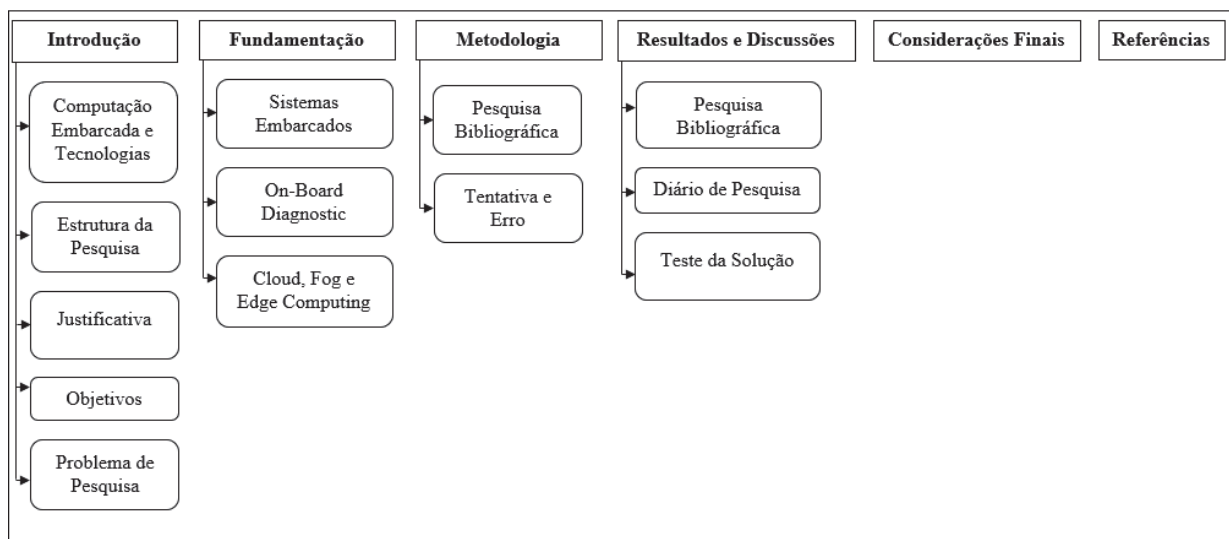
Outro aplicativo muito utilizado para o diagnóstico de veículos – possui de cinco a seis milhões de instalações, em dispositivos móveis com o sistema operacional Android – é o

aplicativo Torque. Produzido em duas versões, sendo uma gratuita e outra paga: Torque Lite e Torque Pro, respectivamente, o aplicativo possui hoje o maior número de usuários no segmento, na loja do Google Play de aplicativos (PORTAL GOOGLE PLAY, 2018).

Com a aplicação Torque, o usuário pode conectar, via Bluetooth, seu dispositivo móvel com um dispositivo de leitura de bordo, que esteja ligado à interface OBDII do veículo. Por meio desta conexão é possível: visualizar a temperatura do veículo; acompanhar a velocidade do veículo; acompanhar o nível de emissão de CO₂; gravar as informações de leitura do veículo, assim como faz a caixa preta do avião; buscar pelas falhas encontradas no veículo em um grande banco de dados da aplicação, podendo, desta forma, saber especificamente a que a anomalia se refere; enviar os logs do sistema por e-mail, *Comma-Separated Values* (CSV) ou *Keyhole Markup Language* (KML) e utilizar o aplicativo como um apoio aos sensores internos do veículo, que interagem com o usuário, como sensores do banco (PORTAL GOOGLE PLAY, 2018a).

Para melhor apresentar a evolução da pesquisa, este trabalho está dividido em cinco capítulos, sendo o primeiro capítulo introdutório, seguido pelo capítulo de Fundamentação Teórica. Em seguida são apresentados os capítulos de Metodologia, Resultados e Discussões, Considerações Finais, além das Referências utilizadas, conforme apresenta a Figura 1.

Figura 1 – Estrutura da Pesquisa



Fonte: O autor.

Justificativa

A presente pesquisa foi desenvolvida tendo-se por justificativa o atual cenário da computação embarcada automotiva e sua notabilidade no setor automotivo, sua evolução nos últimos anos, sua representatividade e tendências, aliadas a uma lacuna de estudo encontrada em pesquisa bibliográfica específica, onde, em sua grande maioria, as pesquisas estudadas procuraram explorar meios de efetuar a análise dos dados, diferentemente desta pesquisa, que busca coletar e disponibilizar os dados.

Objetivos

Portanto, a presente pesquisa tem como objetivo geral o desenvolvimento de um meio para coletar, por meio da interface OBDII, os dados de anomalias armazenados nas unidades de controle eletrônico de veículos e armazená-los em nuvem, permitindo que estes sejam resgatados, processados e disponibilizados para eventuais consultas.

O desenvolvimento da pesquisa tem por objetivos específicos:

- Desenvolver artefatos de softwares capazes de cumprir com o objetivo geral proposto.
- Contribuir para a produção tecnológica do Centro Paula Souza com o registro dos softwares desenvolvidos.
- Acrescentar conhecimento acadêmico, relativo aos sistemas OBD/OBM.

Problema de Pesquisa

Desenvolver um artefato de software que permita armazenar em nuvem, os dados de anomalias, coletados por meio da interface OBDII, permitindo que esses dados sejam resgatados, processados e disponibilizados para eventuais consultas posteriores.

1. FUNDAMENTAÇÃO TEÓRICA

Com o avanço dos componentes eletrônicos, surgiram os chamados Sistemas Embarcados, caracterizados como periféricos semelhantes aos computadores, no que diz respeito à capacidade de execução de tarefas e cálculos, porém, responsáveis pela execução de um pequeno conjunto de tarefas, e de forma mais rápida. Um sistema embarcado é formado pelo hardware e por um software embutido neste hardware, cuja função é executar frequentemente o mesmo conjunto de tarefas (EDWARDS, 2003).

Segundo Edwards (2003), os Sistemas Embarcados estão se tornando uma inevitável parte de qualquer produto ou equipamento de qualquer setor, seja ele de telecomunicação, equipamento médico, controle industrial, aplicação doméstica ou em automóveis.

O computador pessoal é uma ferramenta desenvolvida para o uso geral, onde é possível instalar o sistema operacional de sua preferência, conectar diversos tipos de periféricos, utilizar diversas formas de comunicação, como *Universal Serial Bus* (USB), Bluetooth, WiFi e efetuar outras funções. Para isto, o computador é equipado com componentes que permitem a sua utilização para fins gerais.

Diferentemente, um sistema embarcado é desenvolvido exclusivamente para executar determinado conjunto de tarefas e com a maior eficiência possível. Para isto, seus componentes são precisamente construídos, de modo que não haja perda ou sobra de capacidade em nenhum elemento e que não sejam utilizados componentes de forma desnecessária, garantindo assim o melhor processamento das atividades (SHIBU, 2009).

O primeiro sistema embarcado que se reconhece é o *Apollo Guidance Computer* (AGC), desenvolvido pelo laboratório de instrumentos da Universidade de Michigan, para a expedição lunar da espaçonave Apollo, enquanto a primeira produção em massa de sistemas embarcados foi para o computador de orientação do míssil Minuteman-1, em 1961 (SHIBU, 2009).

Muitas tecnologias atuais são praticamente dependentes de sistemas embarcados para seu funcionamento. No setor automotivo a realidade não é diferente, a necessidade de sistemas embarcados ocorre, por exemplo, em alguns aprimoramentos relacionados às funções de segurança e conforto introduzidas em 2010, como a entrada sem chave, os sistemas de controle de suspensão e os novos sistemas de assistência ao condutor (WATZENIG; ARMENGAUD, 2011).

Com o passar do tempo os sistemas embarcados evoluíram. Na medida em que os processadores utilizados em sistemas embarcados se tornaram mais potentes e sofisticados,

tornou-se viável a utilização de sistemas operacionais para auxiliar o desenvolvimento do software embarcado (GALLASSI; MARTINS, 2014).

Dentre as áreas mais atingidas pelos avanços tecnológicos relacionados aos sistemas embarcados, situa-se o setor automotivo. As montadoras estão cada vez mais voltando os seus projetos para essa tecnologia, com a qual conseguem disputar novas fatias no mercado. Isto ocorre devido ao fato dos consumidores mostrarem sempre mais interesse nos benefícios fornecidos pela utilização de sistemas embarcados, como o aumento do desempenho do veículo, ao mesmo tempo em que se reduz o consumo de combustível (GALHARDI, 2014a).

Segundo Leen e Heffernan (2002) citados por Navet et al. (2005), existem unidades centrais de comunicação no veículo – *Electronic Control Unit* (ECU) – responsáveis por gerenciar as informações obtidas pelos sensores. Inicialmente, a troca de informações entre os diferentes sensores do veículo era feita diretamente entre uma unidade de controle e outra, sem nenhuma forma de comunicação específica. Contudo, com o alto nível de dados fluindo pelas unidades de controle do veículo, este tipo de comunicação tornou-se inviável para a evolução do automóvel. A necessidade de mudança levou à criação do conceito de rede veicular e ao desenvolvimento de regras de comunicação, conceitos apresentados com o desenvolvimento da rede/protocolo *Controller Area Network* (CAN), pela empresa Bosch, na década de 1980. A rede de CAN, cujo conceito de troca de informações se assemelha ao de redes de computadores, foi implementada inicialmente nos veículos Mercedes na década de 1990 e é, até os dias atuais, a mais ampla rede de sistemas automotivos.

Segundo Palocz-Andresen (2012) devido à evolução dos dispositivos eletrônicos veiculares, ao grande número de dados sendo obtidos pelos sensores e aos novos protocolos de comunicação que foram desenvolvidos, tornou-se complexo o diagnóstico por completo de um veículo, pela falta de padronização das informações. Foi então que, segundo a legislação dos Estados Unidos, reconheceu-se a necessidade de uma forma de monitoramento das informações do veículo. Consequentemente, no início dos anos de 1990, foi introduzida a interface de comunicação *On-Board Diagnostic* (OBD).

On-Board Diagnostic (OBD) é uma interface de comunicação que transfere para o exterior de um veículo suas informações do sistema interno. A interface foi desenvolvida com o objetivo de padronizar a forma de coleta das informações de sistemas veiculares, como dados de monitoramento dos componentes do veículo. OBD é dividido atualmente em duas vertentes: OBDI e OBDII (PALOCZ-ANDRESEN, 2012).

Para Palocz-Andresen (2012), a interface inicialmente desenvolvida, chamada OBDI após o lançamento da segunda versão, foi configurada de forma diferente para cada tipo de

veículo. Ela apresentava uma padronização dos dados, mas, não na forma de recebê-los. Além disto, a categoria de dados lidos pela primeira versão não era grande, restringindo-se a:

- Monitoramento da emissão de componentes conectados com o controlador (predominantemente diagnóstico de condução elétrica).
- Saída de códigos de anomalias.
- Soluções de falhas debitados para o fabricante.

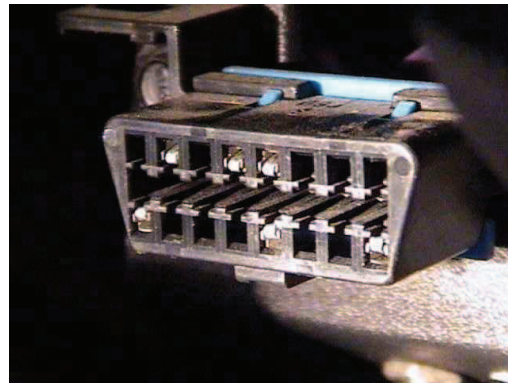
Palocz-Andresen (2012) mostram também a introdução da segunda versão da interface OBD, chamada OBDII, na metade da década de 1990. A nova interface foi lançada com algumas funções extras e com padronização de comunicação dos veículos, sendo as seguintes informações disponibilizadas:

- Monitoramento de todos os sistemas relevantes para o gás de escape e componentes referentes ao mau funcionamento de importantes sistemas de redução de emissões ou de falhas de ignição.
- Armazenamento de falhas com as condições de operação similares.
- Leitura da memória por meio de um dispositivo de saída padronizado.

O monitoramento das operações do veículo é feito a cada certo intervalo de tempo pelo OBD, utilizando para isto a combinação de computadores de bordo, sensores, atuadores, fios, plugues e outros meios similares. Desta forma, é possível identificar anomalias armazenadas na memória interna veicular (PALOCZ-ANDRESEN, 2012).

Segundo Husni et al. (2016), a interface OBDII é comumente posicionada em um local de fácil acesso ao motorista, geralmente localizada abaixo do volante, e se assemelha à interface apresentada na Figura 2.

Figura 2 – Foto de uma Saída OBDII



Fonte: Hack a Day (2017)

Husni et al. (2016) explicam que para que a coleta de dados do OBD seja realizada, é necessário que um dispositivo leitor OBD, capaz de efetuar a leitura das informações e enviá-las por algum meio de comunicação, seja conectado à interface OBD. Para sua pesquisa, os autores utilizaram o dispositivo ELM327, capaz de emitir as informações via Bluetooth. Após a conexão do dispositivo à interface OBD, o dado que se deseja obter da interface deve ser solicitado ao ELM327, por meio de seu código de identificação na interface OBD, chamado *Parameter ID* (PID).

Figura 3 – Exemplo de ELM327



Fonte: Husni et al. (2016).

Os PIDs são códigos, geralmente hexadecimais, utilizados para se solicitar dados à interface OBD. O envio de comandos com PIDs ao OBD é dividido em duas partes, sendo a primeira para representar o modo (tipo de informação a ser consultada) e a segunda para representar a identificação do que será buscado. Existem distintos modos OBD, sendo cada um

responsável por um tipo de informação. Dentro de alguns destes modos existem os PIDs, que são os códigos que representam a informação exata a ser solicitada (HUSNI et al., 2016).

Segundo Dzhelekarski e Alexiev (2005), existem nove modos OBD – chamados pelos autores de *Service* – dos quais nem sempre são todos utilizados, variando de acordo com o protocolo de comunicação do veículo. Na pesquisa os autores apresentam cada um dos *Services* que compõem o protocolo KWP2000 e quais tipos de dados fazem parte deles:

- *Service* \$01: requisição de valores consultados no veículo em tempo real.
- *Service* \$02: requisição de valores armazenados no veículo a cada período de tempo, chamados “dados congelados”.
- *Service* \$03: obtenção de códigos de falhas do sistema, armazenados nas unidades de controle do veículo.
- *Service* \$04: utilizado para apagar as informações armazenadas no sistema.
- *Service* \$07: obtenção de códigos de falhas do sistema ocorridos recentemente, como na última utilização do veículo, chamados “pendentes”.
- *Service* \$09: requisição do número de identificação do veículo.

Segundo Leen e Heffernan (2002), as responsáveis pelo armazenamento e gerenciamento das informações no veículo, são as unidades controle eletrônico (ECU). Ao se solicitar algum dado à interface OBD do veículo, esta interface busca por informações nestas unidades de controle e as retorna para o solicitante (NAVET et al., 2005).

O OBDII fornece uma interface de comunicação padronizada para a coleta de dados do sistema do veículo, ou seja, por meio de uma única interface é possível solicitar todos os dados veiculares disponíveis, porém, a rede de comunicação veicular existente no sistema do veículo pode variar, inclusive na forma com que os dados solicitados podem ser entregues. Essa rede de comunicação interna do veículo é chamada protocolo. Os protocolos OBDII mais utilizados atualmente são: SAE J1850 PWM; SAE J1850 VPW; ISO 14230 KWP (2000); ISO 9141 e ISO 15765-4/SAE J2480 CAN (JIE et al., 2010).

Guimarães (2010, p.209) define:

Protocolos de comunicação são meios de transmissão e recepção de dados utilizados para intercomunicar módulos eletrônicos e/ou sensores e atuadores inteligentes equipados com micro controladores e *transceivers*, por exemplo. Existem vários tipos

de protocolos de comunicação, cada qual com suas características técnicas específicas e, portanto, com as suas aplicações mais apropriadas.

Assim como diversos protocolos de comunicação, existem diversos dispositivos capazes de fazer a comunicação com a interface OBD do veículo, alguns são específicos para cada modelo de protocolo e outros são mais abrangentes. Ao efetuar a leitura da interface OBD, cada modelo de dispositivo pode enviá-las por um meio diferente de comunicação, os mais comuns são: Bluetooth, Wi-Fi e *Wide-Band Code-Division Multiple Access* (WCDMA) (BAEK; JANG, 2014).

Segundo Guimarães (2010), quando se refere à leitura da emissão de gases pelo veículo, o OBD não mede diretamente as concentrações dos gases de escape, mas, correlaciona os sinais adequados dos sensores, utilizando algoritmos complexos no microcontrolador do veículo. O autor destaca, ainda, que somente falhas individuais e previsíveis são reconhecidas pelo sensor OBD, em caso de falhas múltiplas e simultâneas a leitura não é efetuada.

Em contraste com o OBD, a tecnologia *On-Board Measurement* (OBM) detecta a saída real de poluentes, contudo, não é capaz de localizar as fontes de anomalias no motor ou no sistema de gases de escape, o que faz necessário a utilização de ambas as tecnologias OBD e OBM. Isto resulta em um sistema de informação acessível e confiável na medição da produção de poluentes, durante a operação diária do veículo (GUIMARÃES, 2010).

A importância da utilização da interface OBD dos veículos pode ser compreendida pela sua utilização por muitos autores em suas pesquisas. Monitoramento do veículo, prevenção de falhas, análise de falhas, segurança do motorista, comportamento do motorista, redução dos dados armazenados, economia de gastos e controle de poluição são algumas das utilizações para os dados coletados da interface OBD.

Zhang et al. (2013) apresentam uma proposta de um sistema baseado na análise por amostragem, chamado *SafeDrive*. A função da aplicação é detectar anomalias no comportamento de um motorista, trabalhando para isto com duas análises: uma local e outra online. A análise local faz a análise dos dados coletados no veículo e, com eles são construídos gráficos do comportamento do motorista. Posteriormente, é feita a análise online, onde os gráficos gerados são comparados aos dados do comportamento de outros motoristas, já coletados e armazenados em nuvem. Desta forma, é possível verificar se o comportamento do motorista está ou não fora da normalidade.

O sistema OBD do veículo foi desenvolvido para monitorar frequentemente a condição do veículo. Ao detectar uma anomalia, o sistema OBD irá acender a lâmpada de indicação de

defeitos no painel do veículo, para notificar o motorista de que deve procurar imediatamente uma pessoa especializada para a correção da falha (JHOU et al., 2013).

Para melhorar a utilização do OBD pelo usuário do veículo, Jhou et al. (2013) apresentam um conjunto de softwares, cuja funcionalidade é obter os dados do veículo, por meio de um dispositivo ELM327 conectado à interface OBD, e enviá-las para nuvem. Após a recepção dos dados em nuvem, um sistema baseado em experiência faz a análise das informações obtidas e, quando alguma falha é detectada, a aplicação alerta o motorista de uma maneira mais detalhada, do que apenas uma lâmpada indicadora, apresentando detalhes relacionados à falha detectada.

Figura 4 – Lâmpadas de Indicação de Defeito de Painéis dos Veículos.



Fonte: Jhou et al. (2013)

Ao se coletar dados em tempo real de um veículo, por meio do diagnóstico de bordo, o volume de dados obtidos pode ser grande, em função da frequência da coleta. Buscando a redução do volume de dados a ser armazenado em casos onde a coleta em tempo real é necessária, para a melhor utilização da tecnologia OBD. Athavan e Radhika (2015) apresentam em sua pesquisa uma forma diferente de armazenamento: por meio da divisão por faixas. Segundo os autores, a solução para o problema é trocar a forma de armazenamento conhecida como *time-domain* – armazenamento de todos os dados coletados – para o armazenamento *frequency-domain*, onde o armazenamento ocorre por faixas de dados, ou seja, após um período de coleta, que se encerra ao desligar o veículo, é feita a divisão dos dados coletados em faixas e o armazenamento no banco de dados é realizado, gravando-se em cada faixa a sua respectiva quantidade de dados coletados. Desta maneira, os dados são todos gravados de uma única vez e em volumes menores. Nos testes efetuados pelos autores, foi possível a redução de 6.7Gb de dados armazenados, para quarenta bytes.

Amarasinghe et al. (2015) apresentam um sistema, cujo foco é a capacidade de coletar, armazenar e analisar os dados do veículo por um longo período de tempo. Por meio da comunicação via Bluetooth com o dispositivo ELM327 – responsável por coletar e transmitir

as informações do OBD do veículo – o aplicativo desenvolvido pelos autores obtém os dados de diagnóstico. Para tal feito, o aplicativo faz requisições ao ELM327 enviando o modo de informações desejado e qual o parâmetro de identificação da informação também desejada (PID), por exemplo: ao se consultar a temperatura do óleo do motor, a aplicação envia ao ELM327 o comando referente ao modo **01** – bloco de informações que contém a informação relacionada à temperatura do óleo – e o PID **5C**, que representa qual a informação buscada dentro do modo **01**. Assim, o comando enviado ao dispositivo ELM327 é **015C**.

O sistema apresentado por Amarasinghe et al. (2015) é capaz de efetuar, por meio da análise dos dados obtidos pelo OBD do veículo, o monitoramento do motorista e o diagnóstico de veículo. Assim, o sistema é capaz de fornecer ao motorista os alertas relacionados às falhas no veículo, além da indicação de comportamentos inadequados do motorista, que podem resultar em falhas no veículo. Para a análise de comportamentos inadequados, o sistema faz uso de um histórico de dados coletados de outros veículos; e, comparando-se um conjunto de dados de comportamentos de diferentes motoristas e o conjunto de falhas detectadas é possível encontrar relações que permitam identificar antecipadamente as possíveis falhas em função do comportamento do motorista.

Chen et al. (2015) propõem um sistema de leitura e análise veicular para a segurança do motorista. Esse sistema é composto por um leitor OBDII, um banco de dados e um dispositivo móvel com uma aplicação capaz de capturar as informações, analisá-las, alertar o usuário quando necessário e armazenar os dados obtidos no banco de dados que está em nuvem. Ao serem armazenados os dados em nuvem, um profissional pode imediatamente analisar a situação do veículo e prestar o suporte necessário. Em paralelo à análise dos dados, o sistema captura informações de localização do veículo e as armazena, podendo ser feita, por meio da utilização destas informações, a localização de veículos desaparecidos.

A utilização do leitor de bordo, como forma de garantir a segurança do motorista, ainda pode ser aprofundada: uma solução de software é capaz de capturar as informações de veículo e, por meio de análise e/ou introdução de informações pelo usuário da aplicação, detectar obstáculos nas vias; analisar o comportamento do motorista; alertá-lo sobre infrações ou reportar para autoridades caso necessário; e, detectar acidentes e enviar um pedido de socorro para a localidade do veículo, armazenando dados que possam auxiliar na solução do problema (ABUALI, 2015).

Como forma de poupar tempo, com o desenvolvimento de aplicações voltadas para a coleta de dados por meio da interface OBD, ou com o intuito de manter o foco na análise dos dados, não em sua captura, é possível fazer o uso de ferramentas já desenvolvidas como meio

de coleta das informações necessárias. Em sua pesquisa, Hu et al. (2015), fazem uso da aplicação “Torque” como uma estrutura pela qual as informações são avaliadas e enviadas para o armazenamento em nuvem, tais como: consumo de combustível; distâncias percorridas e comportamento do motorista. Segundo apresentam os autores, por meio dessa ferramenta é possível efetuar a leitura dos dados de diagnóstico do veículo e enviá-los para outros locais para que sejam analisados.

Xiong e Gu (2016) utilizam o OBDII para redução de custos de pós-venda. Os autores apresentam uma solução, que coleta as informações dos veículos por meio de um dispositivo leitor dos dados, e os armazena em um banco de dados em nuvem. Após o armazenamento, um sistema faz a leitura dos dados de forma automática e retorna a análise ao usuário, sem a necessidade da interação humana, exceto alguns casos específicos não reconhecidos.

Em acréscimo ao benefício da redução dos custos, o software consegue analisar os dados coletados do veículo e compará-los ao histórico de outras falhas, prevendo assim problemas que possam ser ocasionados pelo mau uso e fornecendo sugestões ao usuário (XIONG; GU, 2016).

Husni et al. (2016) apresentam uma solução para monitoramento do veículo, prevenção de falhas, análise de anomalias e análise do comportamento do motorista. Nessa pesquisa, são coletadas as informações do veículo via OBDII e sua localização, por meio do aplicativo para dispositivos móveis. Após coletadas, as informações são salvas e processadas em um ambiente em nuvem da *International Business Machines* (IBM - IBM BlueMix), que fornece ao usuário respostas às falhas encontradas, previsões de problemas e análise do comportamento do motorista, informações que podem ser úteis para empresas com frotas.

Com o grande número de veículos comercializados atualmente, cresce também a necessidade de softwares com funcionalidades de auxiliar o motorista enquanto dirige. Tais aplicações ajudam o motorista a manter suas atenções voltadas para o ato de dirigir, evitando distrações no trânsito, que possam resultar em acidentes. De acordo com a empresa McKinsey & Company (2016), a perspectiva para o ano de 2030 é de que 15% dos veículos produzidos sejam totalmente autônomos, com isso a segurança no trânsito será maior. A utilização das tecnologias de assistência ao condutor possui mais uma utilidade neste caso: que é fazer com que, aos poucos, os condutores deixem de ter o controle sobre determinadas funções do veículo e, ao mesmo tempo, compreendam a necessidade da segurança no trânsito.

Chou et al. (2017) apresentam uma solução para analisar falhas encontradas e alterações que possam se tornar falhas, ampliando a segurança do motorista. Na solução, os autores utilizam um leitor OBDII, um kit Arduino, LoRa Gateway (wireless de longa distância) e um

banco de dados em nuvem. Ao ser detectada uma anomalia no sistema do veículo, o evento é informado em um display ao motorista e, ao mesmo tempo, o Arduino envia o ocorrido para o banco em nuvem via LoRa. Posteriormente, os dados armazenados são processados e um escopo de manutenção do veículo é construído.

Muitas empresas fazem a utilização de aplicações que se comunicam com a interface OBD dos veículos. Montadoras de veículos e/ou empresas de aluguel de carro, por exemplo, solicitam aos seus clientes que utilizem destes softwares em smartphones para que, ao mesmo tempo em que o cliente mantém um canal de comunicação com a empresa – com ampla disponibilidade de informações relacionadas ao seu produto e sua conta na aplicação – seja possível avaliar informações como: o funcionamento do veículo; trajetos percorridos (por meio do GPS) e comportamento do motorista. Uma preocupação atual, é que devido ao grande número de aplicações capazes de trocar informações relacionadas ao sistema e ao usuário, com outros softwares faz com que aumente a probabilidade do veículo se tornar alvo para ataques cibernéticos (YAN, 2015).

Yan (2015) também desenvolveu uma pesquisa, durante um período de dois anos, cujo objetivo foi o de identificar falhas de segurança nas diferentes formas de comunicação de um veículo. Depois de concluídas as análises, foram encontradas diversas falhas, como a falta de criptografia em sistemas das montadoras, permitindo assim, que pessoas mal-intencionadas capturem os dados de *log-in* da aplicação de um usuário e acessem a sua conta. Empresas de locação de veículos que, por falha na troca de informações entre a aplicação e o servidor, tornam o sistema vulnerável e permitem o acesso alheio ao banco de dados com informações de usuários.

O Quadro 1 apresenta um resumo dos trabalhos encontrados relacionados à atual utilização do OBD.

Quadro 1 – Resumo de Publicações Referentes à Utilização Atual do OBD

<i>SafeDrive Online Driving Anomaly Detection From Large-Scale Vehicle Data</i>	O trabalho apresenta uma proposta de um sistema online, chamado <i>SafeDrive</i> , para detectar anomalias no comportamento de um motorista. As análises são feitas baseadas em informações de outros veículos e motoristas, armazenadas em nuvem.
Zhang et al. (2013) Comportamento do motorista.	
<i>The Implementation of OBD-II Vehicle Diagnosis System Integrated with Cloud Computation Technology</i>	O projeto apresenta um conjunto de softwares capazes de receber as informações do veículo via ELM327 e, por meio de um aplicativo, enviá-las para nuvem. Na nuvem, um sistema baseado em experiência faz a análise das informações obtidas e alerta quando necessário.
Jhou et al. (2013) Monitoramento do veículo; prevenção de falhas; análise de falhas.	
<i>Memory Optimized Lifetime Vehicle Data Acquisition Framework</i>	O trabalho foca uma solução para a redução do tamanho dos dados coletados pelo leitor de bordo. Para isto são armazenadas faixas de valores e a quantidade pertencente a cada uma das faixas. Além da gravação dos dados a solução faz o monitoramento do veículo e repassa as falhas para a montadora.
Athavan e Radhika (2015) Redução dos dados armazenados; monitoramento do veículo; retorno de anomalias à montadora.	
<i>Cloud-Based Driver Monitoring and Vehicle Diagnostic with OBD2 Telematics</i>	Sistema de monitoramento de motorista e diagnóstico de veículo capaz de fornecer alertas ao motorista sobre falhas no veículo; indicar comportamentos inadequados do motorista e fazer a previsão de falhas do veículo de acordo com o histórico de dados, em substituição aos alertas que são apenas fornecidos após falha total do dispositivo.
Amarasinghe et al. (2015) Monitoramento do veículo; prevenção de falhas; análise de falhas; comportamento do motorista.	
<i>An Intelligent Vehicular Telematics Platform for Vehicle Driving Safety Supporting System</i>	Propõem um sistema de captura de análise veicular para segurança do motorista. É composto por um leitor OBDII, um dispositivo mobile com um aplicativo capaz de capturar as informações, alertar o usuário e armazenar os dados em nuvem. O sistema ainda captura e armazena dados de localização do veículo.
Chen et al. (2015) Segurança do motorista; análise de falhas.	
<i>Advanced Vehicular Sensing of Road Artifacts and Driver Behavior.</i>	Apresenta uma solução baseada em um leitor de bordo, um aplicativo para dispositivos móveis e um banco de dados, capaz de capturar as informações de veículo e, por meio de análise e/ou interação com o passageiro, efetuar análises e chamar socorro quando necessário.
AbuAli (2015) Monitoramento do veículo; prevenção de falhas; análise de falhas; comportamento do motorista.	
<i>Towards Mobility-as-a-Service to Promote Smart Transportation.</i>	Refere-se à forma como os dados são sincronizados, e em paralelo a isto, captura os dados veiculares, armazena-os e analisa-os em uma estrutura criada na nuvem.
Hu et al. (2015) Prevenção de falhas; análise de falhas.	
<i>A Remote Engine Health Management System based on Mobile cloud computing</i>	Apresenta um sistema de gerenciamento de funcionamento de motores a <i>diesel</i> . Utiliza dispositivos móveis, capazes de coletar informações em tempo real e armazená-las em nuvem. A análise dos dados permite prever a ocorrência de problemas e fornecer soluções para os já ocorridos.
Xiong e Gu (2016) Monitoramento do veículo; prevenção de falhas.	
<i>Applied Internet of Things (IoT): Car Monitoring System Using IBM BlueMix</i>	Apresenta a construção de um sistema inteligente veicular para monitorar, prever a manutenção e fornecer informações sobre formas eficientes de direção. Os parâmetros monitorados são salvos em nuvem utilizando o IBM BlueMix. A expectativa é que o sistema seja capaz de melhorar a eficiência do veículo por meio da observação e supervisão do comportamento do motorista, permitindo redução da poluição e consumo de combustível.
Husni et al. (2016) Internet das coisas aplicada (IoT): Sistema de Monitoramento de Carro Usando o IBM BlueMix	
<i>Automotive Revolution – perspective towards 2030</i>	A pesquisa relata a perspectiva para o setor automotivo para o ano de 2030, apresentando informações principalmente relacionadas à autonomia dos veículos e ao impacto disto no cotidiano.
McKinsey&Company (2016) Monitoramento do veículo; Segurança do motorista.	

<i>i-Car System: A LoRa-based Low Power Wide Area Networks Vehicle Diagnostic System for Driving Safety</i>	Foco em segurança. Utiliza um leitor OBDII, um Arduino LoRa Gateway (<i>wireless</i> de longa distância) e um banco de dados em nuvem. Ao ser detectado um evento anormal, a informação é disponibilizada em um <i>display</i> , e para nuvem via Arduino/LoRa.
Chou et al. (2017) Segurança do motorista; análise de falhas.	
<i>A two-year Survey on Security Challenges in Automotive Threat Landscape</i>	O projeto apresenta uma pesquisa feita durante um período de dois anos cujo objetivo era identificar falhas de segurança nas diferentes formas de comunicação de um veículo.
Yan (2015) Segurança de informações.	

Fonte: O autor.

Um fator importante para a leitura e análise dos dados emitidos pela interface OBDII de um veículo é a integridade das informações obtidas. Quando sistemas efetuam a leitura dos dados, e os armazenam, não é necessária a interação humana senão na utilização do software que fará a leitura, reduzindo, assim, a probabilidade de falhas. Este processo de captura de dados via interface OBDII e armazenamento é comum na troca de informações entre uma montadora de veículos e a concessionária autorizada (CHILIN; QIN, 2011).

A troca de informações relacionadas à manutenção de veículos, entre a concessionária autorizada e o setor de pós-venda de uma montadora, ocorre de diversas formas, de acordo com a estruturação do serviço. Na China, por exemplo, apesar do grande número de veículos em determinadas regiões, o processo da troca de informações entre a concessionária autorizada e o pós-venda ainda ocorre, muitas vezes, de forma inadequada que, como resultado, traz não apenas à insatisfação do cliente, afetado pela lentidão no conserto do veículo, mas, também, pela perda de recursos financeiros e/ou materiais, e ao armazenamento de informações não confiáveis, relacionadas à qualidade dos veículos produzidos (CHILIN; QIN, 2011).

Ainda, segundo Chilin e Qin (2011), com a saturação de veículos em muitas regiões da China, aumentou-se a dificuldade para um bom acompanhamento do veículo no pós-venda. Com isso, surgiram novas oportunidades de negócio, para empresas terceirizadas especializadas em serviços de pós-venda. Para os autores, os serviços de pós-venda agregam valor em toda a cadeia de suprimentos.

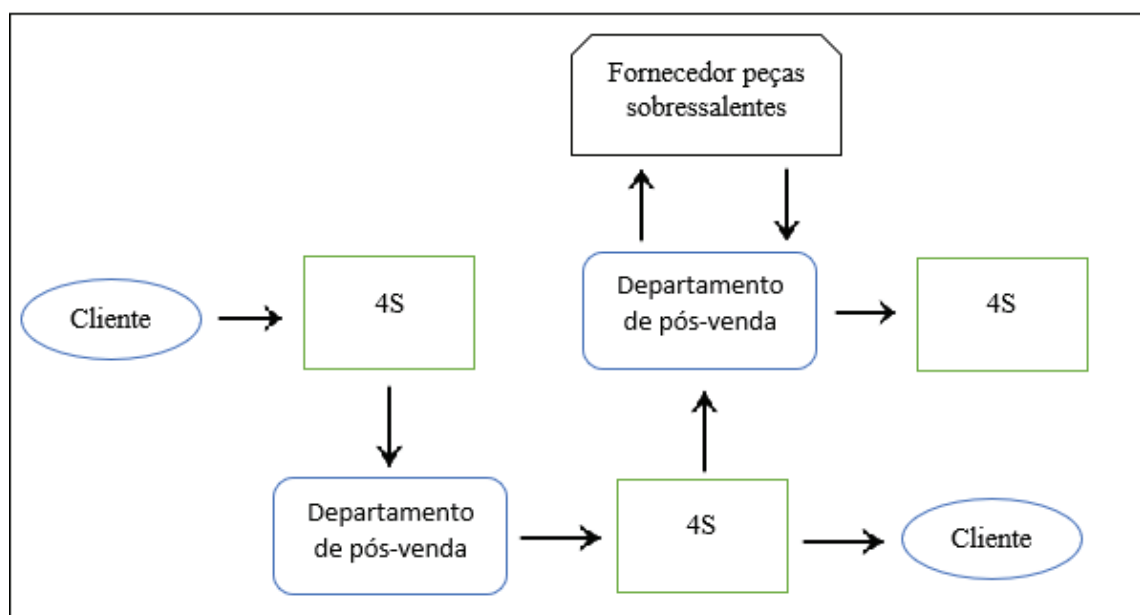
O modelo de pós-venda terceirizado mais conhecido na China é o *Sale, Service, Spare Parts and Survey* (4S). O modelo 4S funciona da seguinte forma: o cliente compra um veículo. Ao necessitar acionar uma oficina autorizada, o cliente procura por uma empresa autorizada do modelo 4S para a montadora e solicita um serviço; a estação 4S faz então uma triagem inicial e encaminha um relatório ao departamento de serviço de pós-venda da montadora, que avalia e aprova a execução do serviço. Após a conclusão do serviço, a autorizada 4S envia um relatório do ocorrido e as peças danificadas para o setor de pós-venda; por fim, o departamento de pós-venda finaliza o serviço, liquida as contas com a autorizada e, ao mesmo tempo, fornecedores

de peças sobressalentes reciclam as peças que estão com a montadora, apresentando um relatório sobre a respectiva qualidade (CHILIN; QIN, 2011).

Apesar de estabelecido o processo, ainda existe a necessidade de aperfeiçoamento do mesmo, uma vez que ocorrem perdas e dificuldades de comunicação no processo, que o torna vagaroso e gera prejuízo (CHILIN; QIN, 2011).

A Figura 5 apresenta o modelo de pós-venda 4S utilizado na China.

Figura 5 – Modelo 4S



Fonte: O autor.

Segundo Chilin e Qin (2011), os principais problemas no modelo utilizado são:

➤ **Perda de Tempo:** a dissociação de informações entre o departamento 4S e a cadeia central da indústria montadora, somada aos dados complexos, levam ao atraso na transmissão das informações e à duplicação de informações, causando perda de tempo no pós-venda. Outro motivo que causa perda de tempo é a falta de cooperação entre indústria e autorizada. O serviço de pós-venda autorizada precisa de um forte suporte das montadoras, quando isso não ocorre, gera lentidão no processo. Ambos os lados necessitam entender adequadamente as necessidades e os dados reais do inventário. A falta desta comunicação e compreensão leva ao desperdício de tempo para a falta de inventário de peças sobressalentes.

Recursos Humanos: diversos modelos de serviços efetuados no pós-venda necessitam suporte de pessoal técnico especializado. A complexidade devido ao grande número de itens de montagem, a falta de padronização e a diversidade de falhas requer dos técnicos de suporte, boas habilidades e conhecimento. Com o desenvolvimento da ciência e da tecnologia, muitos equipamentos avançados de diagnóstico e manutenção estão sendo utilizados no serviço pós-venda, o que demanda por grande quantidade de mão de obra especializada.

➤ Estoques: a coordenação insuficiente entre fornecedores de peças sobressalentes e os serviços de pós-venda têm levado ao aumento significativo do estoque de peças de reposição nos serviços de pós-venda para garantir o fornecimento de peças e assim reduzir o tempo de reparo.

➤ Alto custo e desperdício de recursos: geralmente as estações de serviços de pós-venda, trabalham com todas as marcas e modelos, o que intensifica a dificuldade de controle, e assim demandam por maiores investimentos em logística, software e hardware. A descontinuidade frequente pela falta de peças de reposição causa também descontinuidade no trabalho e constitui fonte de desperdício de recursos.

Para modelos de pós-venda, como o 4S, a possibilidade de montadoras obterem informações relacionadas às falhas ocorridas em seus veículos, de forma rápida e de qualquer local com acesso à internet, é de grande valia, pois permite um feedback mais rápido em relação às anomalias ocorridas em seus produtos. O modelo de pós-venda 4S permite, ainda, que mesmo após o término da garantia, a manutenção dos veículos possa ser executada nas autorizadas, com peças de reposição originais, e as informações relativas às falhas continuem sendo direcionadas às montadoras, para que assim possam melhorar seus veículos (CHILIN; QIN, 2011).

No Brasil existem diversos modelos de comunicação entre concessionária e montadora. A Fiat, uma das principais montadoras atuantes no Brasil, possui um sistema de comunicação próprio, onde o foco é a satisfação do cliente, para que este seja fidelizado e procure a concessionária não apenas para comprar o veículo, mas, sim, para efetuar manutenções (NOGUEIRA, 2002).

Por ter o foco na satisfação do cliente, a Fiat construiu seu modelo de atendimento e treinamento de pessoal interno baseado na velocidade e qualidade como forma de satisfação, buscando, de forma rápida, agendar a visita do cliente, recepcioná-lo, atendê-lo, diagnosticar o problema do veículo, reparar o veículo, verificar se o reparo foi feito adequadamente e entregar o veículo (NOGUEIRA, 2002).

No modelo Fiat, além da rápida comunicação com a montadora, outros dois pontos podem ser destacados por serem diferentes do modelo 4S chinês: autonomia em efetuar a manutenção de veículos sem necessitar aguardar o retorno da montadora - 70% dos veículos devem ser devolvidos no mesmo dia ao cliente; 20% devem ser devolvidos no dia seguinte e 10% em até três dias. Ao término de uma manutenção as peças são entregues ao cliente ou descartadas na frente dele para que não haja reciclagem das mesmas (NOGUEIRA, 2002).

É possível notar que muitos autores utilizam a tecnologia OBD em comunhão com outras tecnologias para o melhor aproveitamento da ferramenta. Smartphones; interfaces de software; GPS; bancos de dados e softwares de análise baseados em experiência são algumas das tecnologias que fazem parte das aplicações com foco na análise dos dados obtidos com o diagnóstico de bordo. Em conjunto com todas as tecnologias citadas, existe outra que está conectada a todas elas, fazendo com que os dados capturados sejam armazenados e disponibilizados de forma rápida e com economia de custos, denominada como *Cloud Computing*.

Cloud Computing ou Computação em Nuvem, segundo Kim (2009), é a utilização de recursos computacionais por meio da Internet, de forma mais econômica, rápida e acessível em relação à infraestrutura local. Para o autor existem quatro principais vantagens na utilização da computação em nuvem: os fornecedores de serviços de computação em nuvem possuem toda a estrutura computacional necessária, o que dispensa investimento inicial, às vezes alto, com infraestrutura local; os usuários podem aumentar ou diminuir seus recursos computacionais utilizados como: memória, processamento e servidores, de forma fácil e flexível; os custos da utilização de recursos em nuvem são geralmente mais baixos, pois o valor pago é referente apenas aos recursos utilizados e tempo de utilização; e, por último, os recursos armazenados em nuvem são acessíveis de qualquer local com conexão à Internet.

A computação em nuvem é um modelo de serviço da tecnologia da informação onde outros serviços (tanto hardware como software) são entregues sob demanda aos usuários por meio da Internet, independente do dispositivo e da localização. Os recursos necessários para fornecer a qualidade do nível de serviço são compartilhados, hábeis para serem ampliados de forma dinâmica conforme a necessidade, rapidamente provisionados, virtualizados e liberados, com a interação mínima dos provedores de serviços. Os usuários pagam pelo serviço como uma despesa operacional, sem qualquer despesa significativa de capital inicial (MARSTON et al., 2010).

Segundo Marston et al. (2010), a computação em nuvem oferece cinco principais vantagens ao usuário:

1. Reduz drasticamente o investimento inicial com tecnologia de empresas menores, que tentam utilizar as vantagens comerciais da computação como ferramenta para análise de negócio.
2. Fornece acesso quase que imediato aos recursos de hardware, tornando o processo inicial de negócio mais rápido.
3. Pode reduzir as dificuldades que impedem a TI de inovar.
4. Torna mais fácil o escalonamento de serviços, sem a necessidade do uso de informações cada vez mais precisas.
5. Possibilita novas classes de aplicações que antes não eram possíveis, como as que respondem em tempo real as informações fornecidas pelo usuário. Um exemplo desta classe de aplicações é a consulta da previsão do tempo.

De acordo com Whaiduzzaman et al. (2013), existem diferentes modelos de entrega do serviço de computação em nuvem, tendo como ponto de vista a visão do usuário final, sendo: *Infrastructure as a Service*; *Platform as a Service* e *Software as a Service*.

Infrastructure as a Service (IaaS) é o modelo onde ocorrem as chamadas virtualizações. Recursos de processamento, rede, hardware e armazenamento estão inclusos neste modelo. Nesta estrutura, dispositivos de infraestrutura e hardware são virtualizados e fornecidos como serviço para que os usuários possam instalar seus sistemas operacionais e trabalhar com suas aplicações (WHAIDUZZAMAN et al., 2013).

Platform as a Service (PaaS) inclui recursos como: sistemas operacionais; gerenciadores de bancos de dados e softwares de gerenciamento de informações. Neste modelo de serviço, os usuários primários são geralmente desenvolvedores de aplicações, pois o ambiente contém capacidade para armazenamento, design de programação e sistema de gerenciamento de outros arquivos (WHAIDUZZAMAN et al., 2013).

Software as a Service (SaaS) entrega softwares e interfaces para clientes, onde o usuário pode utilizar o seu navegador de Internet para acessar a aplicação, e paga de acordo com o serviço consumido. Análises, interação e navegação fazem parte deste modelo (WHAIDUZZAMAN et al., 2013).

Os resultados obtidos com a utilização da computação em nuvem têm se mostrado satisfatórios o suficiente para o surgimento de pesquisas relacionadas à computação veicular em nuvem veicular, para compartilhar o processamento de computadores veiculares. Essas pesquisas buscam utilizar o processamento veicular em tempo ocioso, enquanto estão parados em estacionamentos ou no trânsito. Para isso, enquanto o veículo está parado em algum local, a sua capacidade de processamento estará sendo compartilhada via Internet, desta forma, autoridades do trânsito, por exemplo, poderão utilizar a capacidade de processamento dos computadores de bordo para efetuar simulações de tráfego (WHAIDUZZAMAN et al., 2013).

Muitas vezes o termo computação em nuvem é utilizado para se referir a um modelo de negócio em que os recursos de computação são uma “utilidade”, tal que infraestrutura, armazenamento de dados e processamento de dados são geralmente terceirizados para locais externos, em nuvem. No entanto, computação em nuvem também se refere a um projeto que permite o processamento massivamente escalonável de dados em redes de hardware de baixo custo, como o sistema de arquivos do Google (MCGLOTHLIN; KHAN, 2011).

A computação em nuvem está crescendo rapidamente em popularidade, como solução para o processamento e recuperação de grandes quantidades de dados com custo reduzido. Em setores como o da telecomunicação, os sistemas analíticos têm sido muito afetados, principalmente por consequência dos avanços tecnológicos em smartphones, que deixaram de ser apenas telefones móveis e passaram a conter aplicações, acesso à Internet e grandes capacidades de processamento, produzindo, assim, grandes volumes de dados (MCGLOTHLIN; KHAN, 2011).

Em seu estudo, McGlothlin e Khan (2011) fazem análise de diferentes bancos de dados em nuvem, buscando explorar quais são suas características e qual deles apresenta a melhor eficiência para a recuperação de dados em uma pesquisa analítica. Segundo os autores, devido ao grande volume de dados existentes hoje, é necessário que haja uma estrutura adequada de armazenamento e de consulta, para que a performance dos softwares de análise de dados não seja muito afetada. Para a avaliação, os autores identificaram dez bancos de dados não relacionais comuns de uso no mercado, separando deste grupo um total de três bancos para efetuar os testes de performance, sendo os bancos: Hive, HBase e Cassandra. Após a avaliação dos bancos de dados, os autores apontaram o banco de dados Cassandra como o de melhor performance nas consultas efetuadas.

McGlothlin e Khan (2011) apresentam em seu trabalho a existência de aplicações capazes de gerenciar armazenamento de dados, consultas e análises para que a performance das aplicações possa ser melhorada.

A computação em nuvem é uma alternativa eficiente para que clientes que possuem aplicações Web possam possuir e gerenciar *data centers* privados. Existem diversas vantagens no uso da computação em nuvem, como: maior previsibilidade; localização conveniente, podendo esta ter menor custo; menor custo de manutenção e a capacidade de iniciar o trabalho sem que seja necessário efetuar algumas configurações básicas dos equipamentos. Com o avanço da tecnologia em nuvem, grandes volumes de dados são armazenados diariamente em *data centers* de diversos locais do mundo. Esse excessivo volume de dados, faz com que a velocidade do processamento em nuvem possa sofrer alterações, causando uma pequena latência no retorno de uma requisição, se tornando inviável para determinados fins (BONOMI et al., 2012).

Assim como a computação em nuvem, outros meios de tecnologia têm passado por períodos de desenvolvimento tecnológico. Veículos, computadores pessoais, televisores, dispositivos móveis, casas inteligentes e outros itens comuns de nosso cotidiano passaram a adquirir altas capacidades de processamento, tornando-se capazes de processar informações e prover um retorno, de forma mais rápida que processadores em nuvem. Esses itens estão iniciando um novo conceito de computação, a chamada computação de borda - *Edge Computing* - ou computação em neblina - *Fog Computing* (BONOMI et al., 2012), sendo reconhecidos como dispositivos de borda, por estarem localizados à borda da rede principal de comunicação com o servidor em nuvem, ou seja, às margens da comunicação.

O conceito de *Fog Computing* surgiu por ser um modelo de computação em nuvem com estrutura de rede local virtualizada, entre dispositivos físicos próximos uns aos outros, ou seja, é um modelo em nuvem próximo ao chão. Tal modelo permite: baixa latência no retorno; heterogeneidade; suporte para interações em tempo real e grande número de dispositivos na rede (BONOMI et al., 2012).

A utilização de *Fog Computing* está voltada para situações onde a resposta do processamento deve ser rápida. Ela permite que diversos dispositivos comuns do cotidiano, com capacidade de processamento, interajam entre si por meio de recursos de uma rede de rádio. Como exemplo tem-se um sistema de semáforo, onde o dispositivo interage com diversos sensores a sua volta, os quais detectam a presença de pedestres e bicicletas e calculam a distância e a velocidade dos veículos que estão na via. Com base nessas informações, o sistema do semáforo pode enviar sinais de alerta aos condutores dos veículos que se aproximam ou até alterar seu ciclo de funcionamento para prevenir acidente (BONOMI et al., 2012).

Com o modelo de *Fog Computing*, o retorno de informações aos dispositivos da rede ocorre de forma mais rápida e, ao mesmo tempo, são enviados para o armazenamento em nuvem

apenas os dados necessários, reduzindo assim o volume de armazenamento (BONOMI et al., 2012).

Segundo Hu et al. (2015a), a intenção da *Fog Computing* é desenvolver condições de mercado favoráveis, que criem negócios sustentáveis para todos os participantes da cadeia de valor e facilitem o crescimento do mercado global.

Uma ampla variedade de casos de uso pode ser suportada por mercados novos e inovadores, como: e-health; veículos conectados; automação industrial e realidade aumentada. No que diz respeito a veículos conectados, a comunicação entre os veículos, os sensores da via e uma unidade de gerenciamento, destina-se a aumentar a segurança, eficiência e conveniência do sistema de transporte, pela troca de dados críticos de segurança e operacionais. A comunicação também pode ser usada para fornecer serviços de valor agregado, como localizador de carros, local de estacionamento e suporte a serviços de entretenimento (por exemplo, distribuição de vídeo). O *Edge Computing* pode ser usado para estender a nuvem do carro conectada ao ambiente de estação de base móvel e permitir que dados e aplicativos sejam alojados próximos aos veículos. Isso pode reduzir o tempo de ida e volta dos dados e permitir uma camada de abstração da rede principal e dos aplicativos fornecidos pela Internet (HU et al., 2015a).

Segundo Segura et al. (2017), o grande fluxo de dados gerados por dispositivos móveis pode causar grande tráfego na rede, resultando em redução da performance de aplicações que fazem uso de tecnologias em nuvem. Como forma de explorar o desenvolvimento de uma solução capaz de prover recursos computacionais com baixa latência e com grande mobilidade, reduzindo a necessidade de se utilizar a computação em nuvem, os autores construíram uma arquitetura, formada por dois níveis, onde por meio de *Fog Computing* é possível utilizar dispositivos móveis como recursos computacionais.

No estudo de caso, Segura et al. (2017) trabalham com dois níveis de redes, sendo o nível 1 onde estão localizados os dispositivos móveis e o nível 0, o responsável por gerenciar os níveis de número 1. O nível 1 representa uma rede local, similar à de uma universidade, onde diversos dispositivos móveis estão conectados a um ponto de acesso e a um *cluster* principal, responsável por gerenciar estes dispositivos. No nível 0 – chamado nível mais alto – existe um *cluster* principal, responsável por interligar cada uma das redes locais existentes (nível 1). O fluxo ocorre do nível mais alto para o mais baixo. O *cluster* de nível 0 informa aos *clusters* de nível 1 quais são as tarefas que necessitam ser executadas. Cada *cluster* de nível 1 avalia os dispositivos que nele estão interligados e responde informando se possui ou não capacidade para processar as tarefas requisitadas, dentro do período de tempo estimado e com as

características de qualidade mínimas solicitadas. Caso tenha capacidade para processar uma solicitação, o *cluster* responsável pelo nível mais baixo deve distribuir as tarefas entre seus dispositivos.

Com o desenvolvimento de tecnologias para processamento sob demanda por dispositivos externos, como *Cloud Computing* e *Fog* ou *Edge Computing*, novos negócios estão surgindo e mercados já existentes estão sendo aperfeiçoados de acordo com a necessidade. Cada modelo de negócio possui sua necessidade, sendo imprescindível avaliar a correta tecnologia que nele deve ser utilizada. Variáveis como baixa ou alta latência; heterogeneidade; capacidade de processamento; volume de dados armazenados e número de dispositivos conectados à rede podem variar de acordo com a necessidade. Para o desenvolvimento desta pesquisa, por exemplo, a utilização da computação de borda não foi empregada, pois suas características são incondizentes com o que se precisa, ou seja: o volume de dados armazenados é baixo, pois é enviado para o banco de dados apenas o necessário; não existe comunicação entre diversos dispositivos (apenas entre o leitor de bordo e o computador que está requisitando os dados); a latência do retorno é praticamente irrelevante; não existe rede local virtualizada; o nível do processamento é baixo e não há a necessidade de outros sensores.

2. METODOLOGIA

Inicialmente, uma pesquisa bibliográfica foi conduzida, no sentido de identificar o estado atual da arte em relação ao OBD. Os primeiros estudos apontaram a necessidade de se apreender conhecimentos sobre a interface OBDII, para obtenção de dados de anomalias veiculares, com foco em sua disponibilização para o livre acesso.

Como forma de explorar a necessidade apontada pelo estudo, uma solução inicial foi formulada e proposta. Utilizando-se do método de tentativa e erro chegou-se ao resultado de uma solução composta contendo: quatro aplicativos, um banco de dados e um leitor ELM327. Para a composição desta solução, além das publicações referentes ao tema, foram consultados sites e manuais de desenvolvimento de aplicações, profissionais de oficinas mecânicas multimarcas, e profissionais de concessionárias de veículos.

Segundo Aguiar (2011), existe uma vaga verdade de que toda ciência pode apenas trabalhar com conjecturas. Segundo o autor, as conjecturas podem ser substituídas no decorrer do projeto devido às inadequações e, com a crescente sofisticação da ciência, tornou-se problemático explicar como seria possível gerar hipóteses com termos teóricos a partir da observação. O método de hipóteses (hoje chamado hipotético-dedutivo) atraiu muitos cientistas por parecer fazer justiça à criatividade humana.

O método hipotético-dedutivo materializou o modelo de tentativa e erro, dando ampla liberdade à criação de uma hipótese, a qual será sujeitada a testes. O método combina a liberdade na geração de hipóteses com exigências do processo de testes, considerando como suficiente para a confirmação de uma hipótese a verificação das consequências empíricas (AGUIAR, 2011).

Segundo Bei, Chen e Zhang (2013), o método de tentativa e erro é um método básico da resolução de problemas e aquisição de conhecimento. Para os autores, a abordagem prossegue de forma adaptativa, efetuando testes sequencias nas soluções propostas e observando a sua validade. Quando uma solução proposta é considerada válida, o objetivo está cumprido, caso contrário, sinaliza-se um erro, e nova tentativa se procede.

Diferentes áreas de estudo fazem uso do método de tentativa e erro como, por exemplo: economia, ao se criar estratégias de acordo com as reações do mercado; âmbito empresarial, ao readequar a posição que um funcionário ocupa na empresa com base em suas capacidades e na biomedicina, por meio de testes clínicos (BEI; CHEN; ZHANG, 2013).

3. RESULTADOS E DISCUSSÕES

Este capítulo apresenta, inicialmente, os resultados obtidos em consultas de publicações relacionadas ao tema OBDII. Em seguida, é apresentado o diário de pesquisa, detalhando como chegou-se à solução atual e, por fim, um pré-teste de funcionamento do sistema.

3.1 Pesquisa Bibliográfica

Inicialmente uma busca por publicações abrangentes foi feita para que, por meio de análises bibliométricas, fosse possível avaliar o interesse acadêmico e a possibilidade de uma lacuna de conhecimentos, que pudesse ser explorada como tema desta dissertação.

Pesquisas efetuadas em outubro de 2017, foram realizadas relacionando o OBDII com a computação em nuvem e com veículos. Foram feitas consultas com o termo “*On-Board Diagnostic*” nas bases *Association for Computing Machinery* (ACM) e *Institute of Electrical and Electronic Engineers* (IEEE), retornando um total de 445.080 resultados. A Figura 6 apresenta a evolução do número de publicações relacionadas à *On-Board Diagnostic* ao longo dos anos.

Figura 6 – Publicações Relacionadas à *On-Board Diagnostic* por Período



Fonte: O autor.

Analisando-se o número de citações por ano, é possível verificar que o número de publicações relacionadas ao tema *On-Board Diagnostic* é crescente, evidenciando o grande interesse de pesquisadores.

Foram efetuadas consultas com os termos “OBD II” e “*Vehicular*” nas duas bases de dados ACM e IEEE. Com o filtro “data de publicação” foram selecionadas as publicações mais recentes, partindo do ano de 2014 até o mês de outubro de 2017, data a qual a consulta foi realizada.

A consulta na base da ACM retornou onze artigos, dos quais apenas dois apresentavam citações. Em relação à base da IEEE, foram encontrados oito trabalhos, dentre os quais, três foram citados pelo menos uma vez. Após a leitura dos resumos de cada um dos trabalhos, três deles foram separados por apresentarem relevantes informações e/ou aderência para esta pesquisa.

No quesito processamento de informações, a literatura apresenta vários casos com dispositivos móveis como os smartphones. Com relação à coleta de dados do OBD, observou-se na literatura que, nos aplicativos atuais, esses dados estão sendo coletados em conjunto com dados de geolocalização, e outros dados informados pelo usuário do sistema.

Outras duas consultas foram efetuadas nas duas bases em outubro de 2017, utilizando como filtros: “OBD” – “*Cloud*” e “ELM327” – “*Cloud*”. Na primeira consulta foram identificadas 21 publicações, e na segunda não foi obtida nenhuma publicação.

Posteriormente, todas as 21 publicações foram submetidas ao software *EndNote*, para a remoção de trabalhos repetidos. Com isto, quatro documentos foram eliminados. Após a leitura dos *abstracts*, separando-se quais eram efetivamente relacionados ao tema, restaram nove documentos, todos publicados entre 2013 e 2017. Embora tenham sido encontradas pesquisas similares, nenhuma apresentou a possibilidade de disponibilizar dados a outrem, que não os do interior do veículo, ou os específicos para proprietários de frotas.

As publicações analisadas ainda trouxeram à luz, sete outros tópicos importantes: monitoramento do veículo; prevenção de falhas; análise de falhas; segurança das informações; segurança do motorista; comportamento do motorista e redução de dados armazenados.

A análise das publicações selecionadas revelou os seguintes pontos importantes:

- O interesse acadêmico referente ao diagnóstico de bordo tem sido contínuo e crescente, e de maneira mais acentuada a partir do início dos anos 2000.

- Os softwares para diagnóstico de bordo já apresentam grande utilização no gerenciamento de frotas.

➤ Existe a necessidade de se aprofundar e ampliar os estudos relacionados à utilização de leitores de bordo e computação em nuvem.

➤ Existe a possibilidade de ampliar os estudos relacionando informações de desempenho veicular com melhoria contínua, desenvolvimento de produtos, competitividade das marcas, etc.

a. Diário de Pesquisa

Desde a ideia inicial até a conclusão da solução aqui apresentada ocorreram dificuldades e obstáculos, algumas vezes intransponíveis, que conduziram à atual solução. Considera-se relevante apresentar as duas tentativas iniciais de solução:

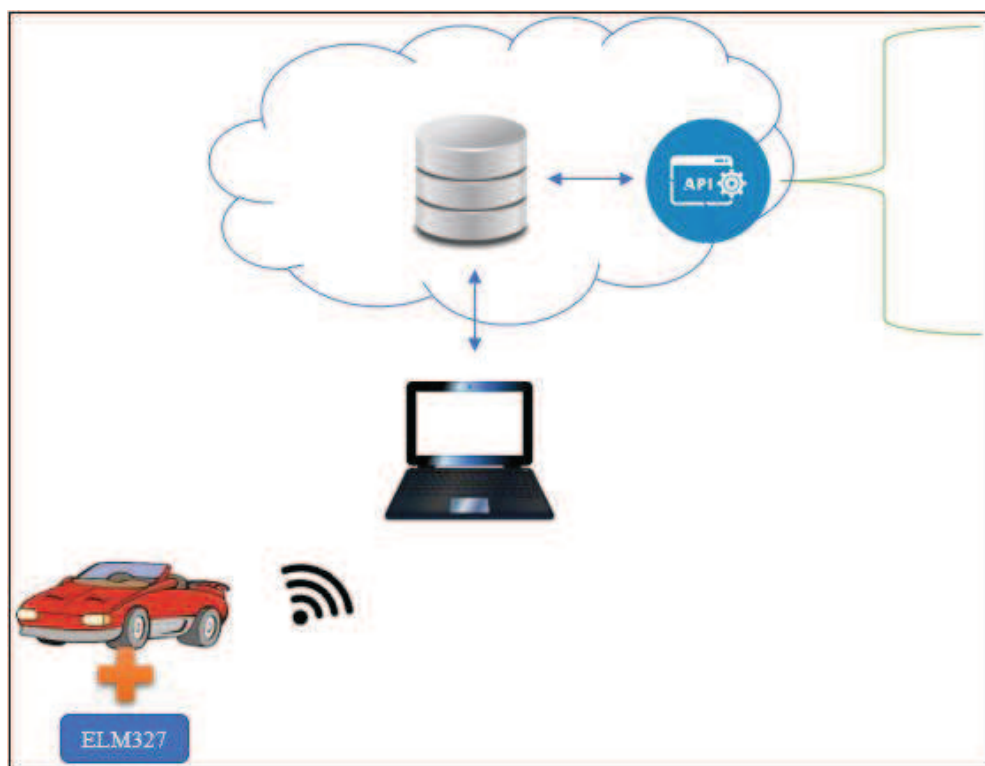
1) Aqui se idealizou a construção de um aplicativo para dispositivos móveis (Android), utilizando o MS Xamarin, com a função de obtenção de códigos de anomalias veiculares, em tempo real de uso. Para isto o motorista deveria ter o aplicativo instalado em um dispositivo móvel e um leitor de bordo ELM327. Após aproximadamente seis meses de programação, esta alternativa foi abandonada pela impossibilidade do aplicativo estabelecer conexão com o Bluetooth do ELM327.

2) Aqui foi desenvolvida uma aplicação Web, pela qual o usuário poderia coletar os dados por qualquer dispositivo capaz de estabelecer conexões com a Internet e com o ELM327. Esta opção se baseou no fato de que uma das tendências futuras em Computação Embarcada Automotiva é a comunicação inter-veicular, o que permitiria o próprio computador de bordo enviar as informações do sistema OBD para uma aplicação Web. Devido à indisponibilidade de recursos, esta tentativa também foi abandonada, após aproximadamente três meses. Isto, principalmente, porque existem as seguintes limitações: o alcance das comunicações via Bluetooth, forma pela qual o dispositivo ELM327 emite os dados coletados, que é de aproximadamente 2,0 m; e quando uma aplicação Web é disponibilizada para acesso em nuvem, a aplicação passa a ser processada em um servidor situado em qualquer lugar do planeta, com uma distância infinitamente superior ao alcance do Bluetooth.

Finalmente, a alternativa concluída foi composta por aplicações instaladas em um computador com sistema operacional Windows, um dispositivo ELM327, um banco de dados e uma interface de programação armazenados em nuvem no Microsoft Azure.

A estrutura da solução desenvolvida foi formada por quatro módulos em uma solução do Microsoft Visual Studio 2013, um banco de dados hospedado no Microsoft Azure, e um dispositivo ELM327. Dentre tais módulos, o *Data* e o *Model* representam as estruturas do banco de dados (*CarLoggerDB*). O terceiro módulo foi o aplicativo utilizado para interpretar os dados do leitor de bordo (*CarLogger*); enquanto o último é uma API (*CarLoggerAPI*), que permite a consulta de dados armazenados.

Figura 7 – Modelo Proposto



Fonte: O autor.

Data e Model: trabalham de forma conjunta no gerenciamento das tabelas do banco de dados.

a) O *Model* possui classes que representam as tabelas do banco de dados com seus respectivos atributos. Utiliza o modelo *Entity Framework*, que possibilita alterar as propriedades e tabelas, sem que seja necessário que o desenvolvedor escreva um novo código SQL (*Structured Query Language*). Ele é o responsável por representar os atributos de uma tabela, ou seja, as informações de cadastro como, por exemplo: **Nome**, **Endereço** e **Telefone** na tabela de **Clientes**.

b) O módulo *Data* é o responsável pela representação da estrutura do banco de dados e suas tabelas, ou seja, quando é necessário buscar informações de uma determinada tabela do banco de dados é o *Data* que é acionado como, por exemplo, as tabelas: **Clientes**, **Veículos** ou **Serviços**.

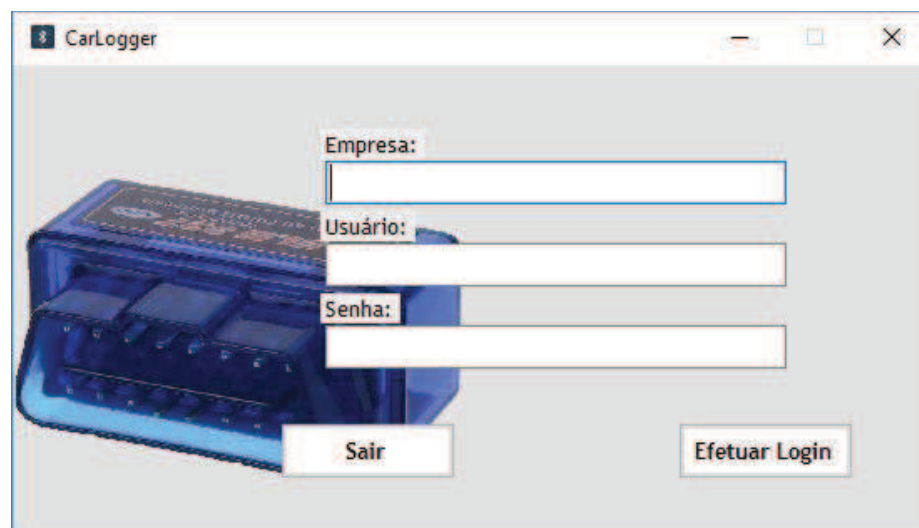
CarLogger: este módulo é o responsável por fazer a comunicação via Bluetooth com o ELM327 e solicitar os códigos de anomalias armazenados na *ECU* do veículo. Após receber os dados solicitados, a aplicação converte os códigos recebidos em hexadecimais, para códigos padrões de anomalias veiculares. Em seguida esses códigos são enviados ao banco de dados na nuvem.

O sistema exige um cadastro prévio do veículo, que aqui abrange os seguintes dados: nome do proprietário; endereço; identificação; telefone; marca, modelo, cor e ano; placa. Assim, ao se capturar os códigos de anomalias, os mesmos serão atrelados a alguns dados cadastrais (somente os do veículo, e não os do proprietário), o que permite consultas aos códigos armazenados pela API.

CarLoggerAPI: este módulo é o responsável por receber as solicitações externas de acesso aos dados, e retorna com as informações de acordo com os filtros disponíveis: modelo, intervalo temporal de captura da informação e código da anomalia.

A Figura 8 representa a tela de login, para acesso ao módulo *CarLogger*.

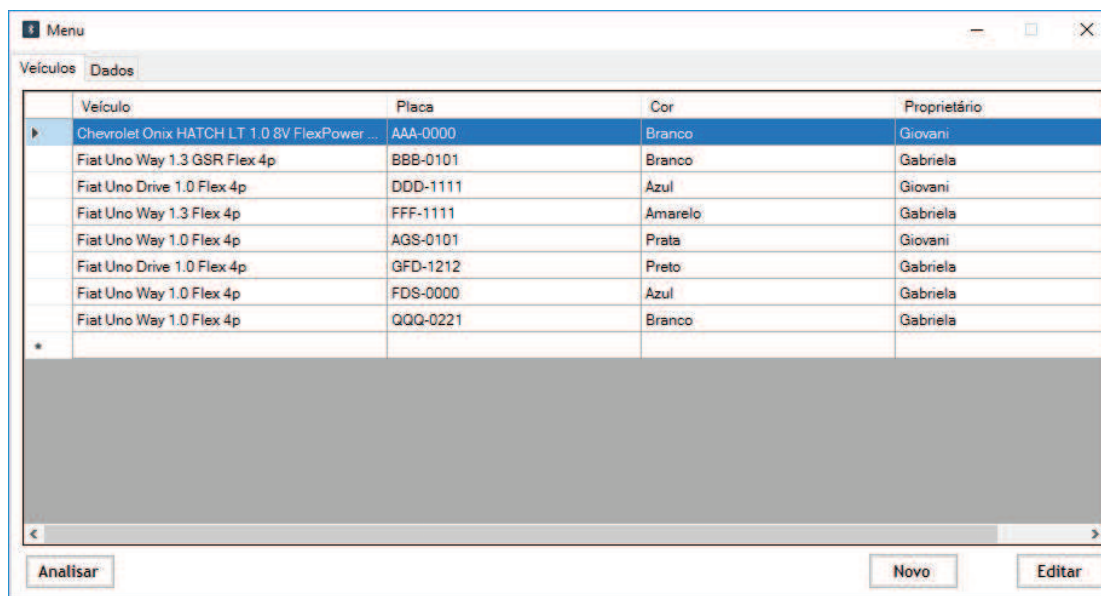
Figura 8 – Tela de Login



Fonte: O autor.

Após a autenticação no sistema, a tela menu apresenta os dados dos veículos, e as opções para edição, inclusão (novo) ou análise, como pode ser visto na Figura 9.

Figura 9 – Menu do Sistema



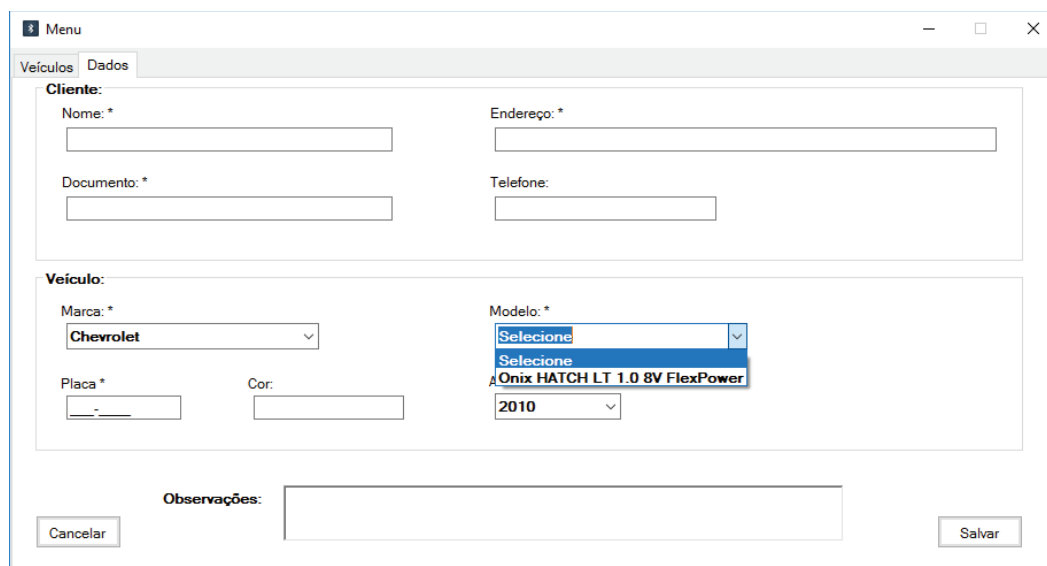
Veículo	Placa	Cor	Proprietário
Chevrolet Onix HATCH LT 1.0 8V FlexPower...	AAA-0000	Branco	Giovani
Fiat Uno Way 1.3 GSR Flex 4p	BBB-0101	Branco	Gabriela
Fiat Uno Drive 1.0 Flex 4p	DDD-1111	Azul	Giovani
Fiat Uno Way 1.3 Flex 4p	FFF-1111	Amarelo	Gabriela
Fiat Uno Way 1.0 Flex 4p	AGS-0101	Prata	Giovani
Fiat Uno Drive 1.0 Flex 4p	GFD-1212	Preto	Gabriela
Fiat Uno Way 1.0 Flex 4p	FDS-0000	Azul	Gabriela
Fiat Uno Way 1.0 Flex 4p	QQQ-0221	Branco	Gabriela

Buttons: Analisar, Novo, Editar

Fonte: O autor.

A Figura 10 apresenta a tela de cadastro, acionada pelos comandos: Novo e Editar.

Figura 10 – Tela de Cadastro



Cliente:

Nome: *

Documento: *

Endereço: *

Telefone:

Veículo:

Marca: *

Modelo: * (dropdown menu open showing: Selecione, Onix HATCH LT 1.0 8V FlexPower, 2010)

Placa: *

Cor:

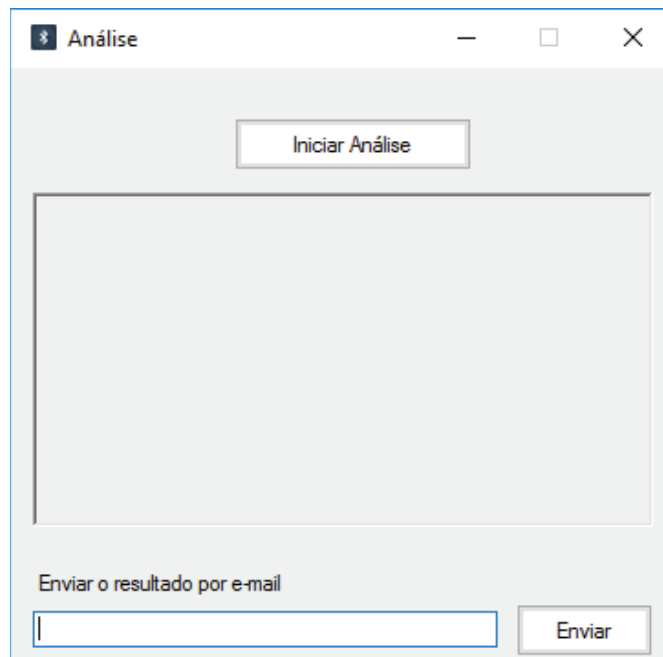
Observações:

Buttons: Cancelar, Salvar

Fonte: O autor.

Na Figura 11 apresenta-se a tela de análise de um item do banco de dados. O comando “Iniciar Análise”, é o responsável por iniciar a comunicação via Bluetooth com o ELM327 e fazer a consulta dos dados.

Figura 11 – Tela de Análise



Fonte: O autor.

Alguns dispositivos ELM327 podem requisitar senha para conexão. Ao iniciar a solicitação, o *CarLogger* se conecta com o dispositivo ELM327, e envia as configurações iniciais para a comunicação.

Para se estabelecer a conexão é necessário inicializar uma das portas de comunicação Bluetooth do computador. A consulta das portas é feita com os seguintes comandos C#:

```
string[] availablePorts = SerialPort.GetPortNames();
```

Identificadas as portas de comunicação do Bluetooth, é constituída uma lista apenas com seus nomes, com as seguintes linhas de comandos:

```
string portList = "";
foreach (String s in ports)
{
    portList += s.ToString() + ", ";
}
```

Um objeto *SerialPort* informa o nome da porta de comunicação (*PortName*), o valor da taxa de transmissão de dados (*BaudRate*), o tempo limite para envio do comando ao dispositivo

ELM327 (*WriteTimeout*) e o tempo limite para recebimento da resposta (*ReadTimeout*); enquanto o comando ***foreach***, testa a inicialização de cada porta a partir destas características, e das seguintes linhas de comandos:

```
foreach (var portName in portList)
{
    try
    {
        SerialPort obdDevice = new SerialPort();
        obdDevice.PortName = portName;
        obdDevice.BaudRate = 38400;
        obdDevice.WriteTimeout = 500;
        obdDevice.ReadTimeout = 500;
        obdDevice.Open();
    }
    catch (Exception ex)
    {
        valid = false;
    }
}
```

Os comandos de inicialização são inseridos em estrutura de ***try catch***. Efetuada a conexão, alguns comandos de configuração são enviados ao ELM327. Inicialmente, o comando ‘AT SP h’ identifica no ELM327 o protocolo para a troca de informações que é específico para o veículo em análise. No exemplo, com o veículo Chevrolet Onix LT 1.0 2013, identificou-se o protocolo padrão de número **seis** do ELM327, que representa ‘ISO 15765-4 CAN (11 bits)’ segundo o documento ELM (2016).

O envio de comandos ao dispositivo ELM327 deve ser feito com a utilização da função ***Write*** da porta de comunicação. Após o envio, é necessário que a aplicação aguarde um período (*delay*) de 255 milésimos de segundo. No exemplo, a configuração do protocolo padrão a ser utilizado é feita pelos comandos:

```
string comando = "AT SP 6";
serialPort.Write(comando);
System.Threading.Thread.Sleep(255);
string resposta = serialPort.ReadExisting();
```

Configurado o protocolo, o ELM327 deve ser reinicializado por meio do comando “AT Z”. Em seguida, o comando “03” é enviado ao ELM327, para a requisição dos códigos de

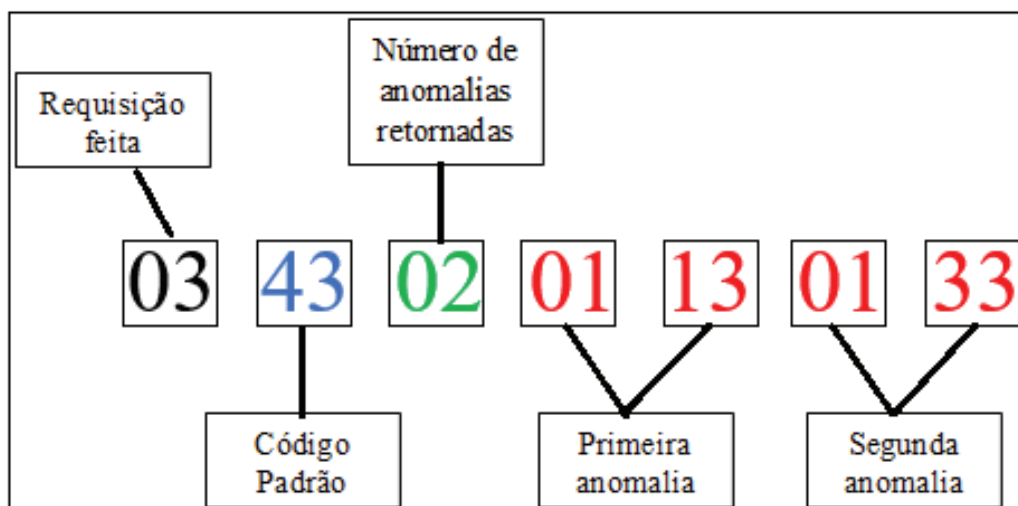
anomalias armazenados no ECU do veículo. O retorno desses dados deve ser decodificado segundo o protocolo adotado. Assim, no exemplo, o protocolo ISO 15765-4 CAN, retorna uma anomalia por meio de caracteres, divididos em pares. Como exemplo, num retorno recebido com “03 43 01 01 13”, os dois caracteres iniciais (03) representam a requisição feita ao dispositivo; o par seguinte (43) representa um código padrão do ELM327; o terceiro par mostra a quantidade de anomalias encontrados (01) e os últimos dois pares o código da anomalia (01 13).

O retorno da solicitação dos códigos de anomalia varia de acordo com o protocolo utilizado pelo veículo. No exemplo anterior, cujo retorno foi: “03 43 01 01 13”, trata-se de um retorno típico de um veículo que utiliza o protocolo CAN.

Inicialmente, o software *CarLogger* foi desenvolvido para interpretar apenas resultados de solicitações feitas a veículos com o protocolo CAN. Após alguns testes efetuados em um Chevrolet Celta 1.0 2012, com códigos de falha reais; ou seja: que não foram ocasionados propositalmente, e pela interpretação de informações encontradas relacionadas ao tema em fóruns online e informações gerais da Internet, notou-se que o protocolo ISO 14230 KWP (2000), que é o utilizado pelo veículo em questão, apresenta retorno próximo ao do CAN e, portanto, factível de interpretação.

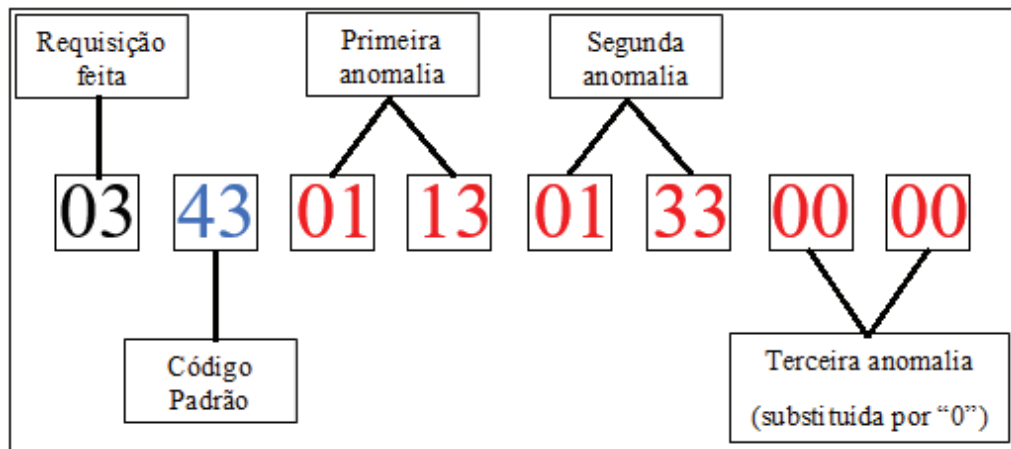
O retorno do protocolo CAN constitui-se de três pares de números informativos, seguidos de outros dois para representar a anomalia. Caso haja mais de uma anomalia, essas serão acrescentadas ao final, sempre em dois pares de números, como apresentado na Figura 12.

Figura 12 – Detalhes do Retorno com Protocolo CAN



Fonte: O autor.

Figura 13 – Detalhes do Retorno com Protocolo KWP



Fonte: O autor.

Já o protocolo KWP apresenta duas diferenças significativas em relação ao protocolo CAN. Primeiro, porque ele não apresenta o terceiro par de números, responsável por indicar as anomalias do veículo. Segundo, porque possui um de tamanho fixo, com dezesseis caracteres, onde estão previstas sempre (e no máximo) três anomalias; estas são substituídas por “00 00” quando não detectadas todas as anomalias do retorno. Por exemplo: o mesmo retorno apresentado na Figura 12, é estruturado no retorno do protocolo KWP como “03 43 01 13 01 33 00 00”. A Figura 13 apresenta com mais detalhes essa estrutura.

A transformação do código de anomalia recebido por meio da comunicação com o OBDII em um código padrão da SAE J2012 é feita por meio de um quadro que utiliza o primeiro dígito do código retornado para identificar onde a falha está localizada no veículo, e que pode ser: *Powertrain (P)*; *Chassis (C)*; *Body (B)* ou *Network (U)*.

Segundo ELM (2016), para a correta conversão do código de retorno em um código padrão SAE J2012, deve-se substituir o primeiro dígito pelo código de localização da falha no veículo. Em seguida, basta-se adicionar os demais dígitos. Por exemplo: o código de retorno “01 13” possui como dígito inicial o “0” que, quando localizado no quadro de conversão, representa “P0”. Acrescentando-se os demais dígitos da anomalia, obtêm-se o código “P0113”, e que representa baixa tensão no sensor devido à variação da temperatura do motor.

O Quadro 2 apresenta os códigos de localização da falha no veículo para conversão do código de retorno do OBD.

Quadro 2 – Caracteres para Localização do Código de Falha

Original	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Equivalente	P0	P1	P2	P3	C0	C1	C2	C3	B0	B1	B2	B3	U0	U1	U2	U3

Fonte: Adaptado de ELM (2016).

Veículos mais novos apresentam um maior número de sensores, assim, emitem com maior frequência alertas de falhas aos motoristas (pelo MIL).

Nesta pesquisa, num veículo investigado, o Renault Logan 2011/2012, o sistema OBDII do veículo não apontou falha, mesmo com o não funcionamento de uma das velas. Presume-se que tal fato ocorreu devido à existência de apenas um único sensor de ignição que verifica a capacidade das velas. Desta forma, com apenas uma delas danificada, os danos causados no sistema de ignição não foram suficientes para acionar o sistema de diagnóstico de bordo do veículo. Ou seja, não acionou a lâmpada MIL, embora o código da falha tenha sido registrado no OBD.

A presença de sensores para verificar precisamente o funcionamento de cada componente do veículo, implicaria em maior número de sensores e cabos conectores, com dois aspectos importantes a serem considerados: o aumento de custo pelo aumento dos eletrônicos embarcados, e o aumento de consumo de combustíveis pelo aumento de peso. Neste contexto, o OBD caminha observando: por um lado as restrições de uso e por outro a evolução tecnológica.

Concluída a etapa de verificação e obtenção do código de anomalias pela aplicação, os dados coletados foram automaticamente enviados para o banco de dados *CarLoggerDB*, de modo a ficarem disponíveis para acesso por meio da API. A solicitação dos dados foi feita por uma requisição GET, informando na *Uniform Resource Locator* (URL) os campos: *Model*; *DTC*; *StartDate* e *EndDate*.

Exemplo:

“*carloggerapi.azurewebsites.net/api/Values/GetLogs?ModelId=7&DTC=P0102&StartDate=&EndDate=*”.

Os campos enviados na requisição executam a função de filtros para busca: o campo *Model* responsável por identificar o modelo do veículo buscado; o campo *DTC* (*Data Trouble Code*) para representar o código da anomalia e os campos *StartDate* e *EndDate* para representar,

respectivamente, a data inicial e a data final de armazenamento dos códigos de anomalia no Banco de Dados.

b. Teste da Solução

Para verificar o funcionamento da solução proposta, foram feitos testes em seis veículos de diferentes marcas. No total foram obtidos dez diferentes códigos de anomalia, dos quais sete destes códigos foram ocasionados propositalmente no veículo e outros três códigos foram coletados em veículos com falhas não propositas, cuja luz de indicação de falha do OBD já estava acessa no painel.

Para ocasionar propositalmente falhas no veículo, foram desconectados sensores que controlam o correto fluxo de ar ou a posição do virabrequim. Ao desconectar um sensor e dar a partida no veículo, o sistema OBD identificou a inatividade do sensor e, conseqüentemente, a falha. O processo acontece da seguinte maneira: ao identificar a falha após a partida, o sistema interno do veículo mantém a falha como um código pendente, ou seja, que está ocorrendo pela primeira vez, sendo assim, o código ainda não foi armazenado na memória. Ao desligar o veículo e tornar a ligá-lo já é possível identificar o código como uma anomalia salva em memória e, então, é possível efetuar a obtenção do código por meio da aplicação *CarLogger*.

Em leituras efetuadas nos veículos Chevrolet Celta e Nissan March, por meio da aplicação *CarLogger*, não foi possível verificar a afirmação feita por Guimarães (2010) sobre a impossibilidade do sistema OBD efetuar a leitura de falhas múltiplas e simultâneas. Durante a validação da aplicação, dois códigos de falhas retornaram para o Chevrolet Celta e três códigos para o Nissan March, simultaneamente, revogando a afirmação feita pelo autor. Possivelmente, com os avanços da tecnologia, essa afirmação já não é mais aplicável.

Nas leituras de falhas efetuadas, foram obtidos códigos de anomalia por meio de dois diferentes protocolos de comunicação: o ISO 14230 KWP (2000) e o ISO 15765-4 CAN.

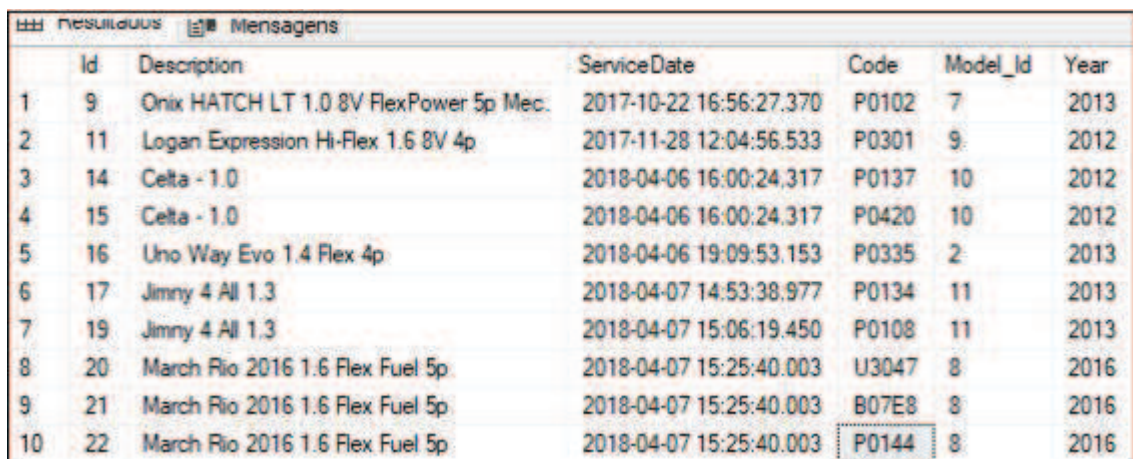
A seguir, no Quadro 3, são apresentadas quais foram as leituras efetuadas e seus respectivos resultados, apresentando também, se o resultado obtido foi proveniente de uma falha simulada ou não.

Quadro 3 – Análises Efetuadas nos Testes da Solução

Veículo	Protocolo	Resultado	Falha Provocada
Chevrolet Onix 1.0 2013	CAN	P0102	Sim
Renault Logan 1.6 2011/2012	CAN	P0301	Sim
Chevrolet Celta 1.0 2012	KWP	P0137	Não
Chevrolet Celta 1.0 2012	KWP	P0420	Não
Fiat Uno Way 1.4 2013	CAN	P0335	Sim
Suzuki Jimny 4 All 1.3 2013	KWP	P0134	Não
Suzuki Jimny 4 All 1.3 2013	KWP	P0108	Sim
March Rio 2016 1.6 Flex Fuel 5p	CAN	U3047	Sim
March Rio 2016 1.6 Flex Fuel 5p	CAN	B07E8	Sim
March Rio 2016 1.6 Flex Fuel 5p	CAN	P0144	Sim

Fonte: O autor.

A Figura 14 apresenta os dados armazenados no banco de dados *CarLoggerDB*, com as características do veículo e do código obtido. A Figura 15 apresenta o retorno da interface *CarLoggerAPI* para a requisição de todos os códigos coletados pela aplicação *CarLogger* e armazenados no banco de dados.

Figura 14 – Resultados Obtidos e Salvos no Banco de Dados


	Id	Description	ServiceDate	Code	Model_Id	Year
1	9	Onix HATCH LT 1.0 8V FlexPower 5p Mec.	2017-10-22 16:56:27.370	P0102	7	2013
2	11	Logan Expression Hi-Flex 1.6 8V 4p	2017-11-28 12:04:56.533	P0301	9	2012
3	14	Celta - 1.0	2018-04-06 16:00:24.317	P0137	10	2012
4	15	Celta - 1.0	2018-04-06 16:00:24.317	P0420	10	2012
5	16	Uno Way Evo 1.4 Flex 4p	2018-04-06 19:09:53.153	P0335	2	2013
6	17	Jimny 4 All 1.3	2018-04-07 14:53:38.977	P0134	11	2013
7	19	Jimny 4 All 1.3	2018-04-07 15:06:19.450	P0108	11	2013
8	20	March Rio 2016 1.6 Flex Fuel 5p	2018-04-07 15:25:40.003	U3047	8	2016
9	21	March Rio 2016 1.6 Flex Fuel 5p	2018-04-07 15:25:40.003	B07E8	8	2016
10	22	March Rio 2016 1.6 Flex Fuel 5p	2018-04-07 15:25:40.003	P0144	8	2016

Fonte: Tela do CarloggerDB.

Figura 15 – Resultados Obtidos Por Meio do *CarLoggerAPI*

```

<LogReturn>
  <Automaker>Chevrolet</Automaker>
  <DTC>P0102</DTC>
  <Date>2017-10-22</Date>
  <Model>Onix HATCH LT 1.0 8V FlexPower 5p Mec.</Model>
  <ModelId>7</ModelId>
  <Year>2013</Year>
</LogReturn>
<LogReturn>
  <Automaker>Renault</Automaker>
  <DTC>P0301</DTC>
  <Date>2017-11-28</Date>
  <Model>Logan Expression Hi-Flex 1.6 8V 4p</Model>
  <ModelId>9</ModelId>
  <Year>2012</Year>
</LogReturn>
<LogReturn>
  <Automaker>Chevrolet</Automaker>
  <DTC>P0137</DTC>
  <Date>2018-4-6</Date>
  <Model>Celta - 1.0</Model>
  <ModelId>10</ModelId>
  <Year>2012</Year>
</LogReturn>
<LogReturn>
  <Automaker>Chevrolet</Automaker>
  <DTC>P0420</DTC>
  <Date>2018-4-6</Date>
  <Model>Celta - 1.0</Model>
  <ModelId>10</ModelId>
  <Year>2012</Year>
</LogReturn>
<LogReturn>
  <Automaker>Fiat</Automaker>
  <DTC>P0335</DTC>
  <Date>2018-4-6</Date>
  <Model>Uno Way Evo 1.4 Flex 4p</Model>
  <ModelId>2</ModelId>
  <Year>2013</Year>
</LogReturn>
<LogReturn>
  <Automaker>Suzuki</Automaker>
  <DTC>P0134</DTC>
  <Date>2018-4-7</Date>
  <Model>Jimny 4 All 1.3</Model>
  <ModelId>11</ModelId>
  <Year>2013</Year>
</LogReturn>
<LogReturn>
  <Automaker>Suzuki</Automaker>
  <DTC>P0108</DTC>
  <Date>2018-4-7</Date>
  <Model>Jimny 4 All 1.3</Model>
  <ModelId>11</ModelId>
  <Year>2013</Year>
</LogReturn>
<LogReturn>
  <Automaker>Nissan</Automaker>
  <DTC>U3047</DTC>
  <Date>2018-4-7</Date>
  <Model>March Rio 2016 1.6 Flex Fuel 5p</Model>
  <ModelId>8</ModelId>
  <Year>2016</Year>
</LogReturn>
<LogReturn>
  <Automaker>Nissan</Automaker>
  <DTC>B07E8</DTC>
  <Date>2018-4-7</Date>
  <Model>March Rio 2016 1.6 Flex Fuel 5p</Model>
  <ModelId>8</ModelId>
  <Year>2016</Year>
</LogReturn>
<LogReturn>
  <Automaker>Nissan</Automaker>
  <DTC>P0144</DTC>
  <Date>2018-4-7</Date>
  <Model>March Rio 2016 1.6 Flex Fuel 5p</Model>
  <ModelId>8</ModelId>
  <Year>2016</Year>
</LogReturn>

```

Fonte: Tela do *CarLoggerAPI*.

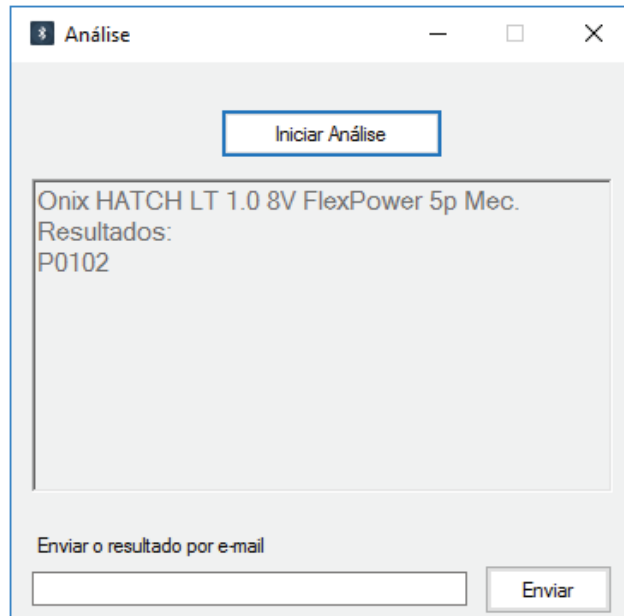
O funcionamento do sistema pode ser melhor compreendido com a descrição dos resultados obtidos com o veículo Chevrolet Onix 1.0 - 2013, que apresenta o protocolo de comunicação 15765-4 (CAN 11/500):

1) Numa primeira análise nenhum código de anomalia foi encontrado, ou que o veículo apresentava alguma falha. Por este motivo foi necessário gerar uma anomalia para que pudesse ser capturada pela aplicação e acusar seu funcionamento, então, o sensor responsável pela análise do fluxo de ar do motor foi desconectado.

2) Após o sensor ter sido desconectado, o código referente à anomalia no fluxo de ar foi capturado pelo OBDII do veículo e um sinal de alerta foi exibido no painel. Em seguida, por meio da utilização da aplicação *CarLogger* e do dispositivo ELM327, foi possível obter o código de anomalia armazenado no sistema interno do veículo. Os dados obtidos foram então exibidos no campo de texto da aplicação e enviados para o e-mail informado.

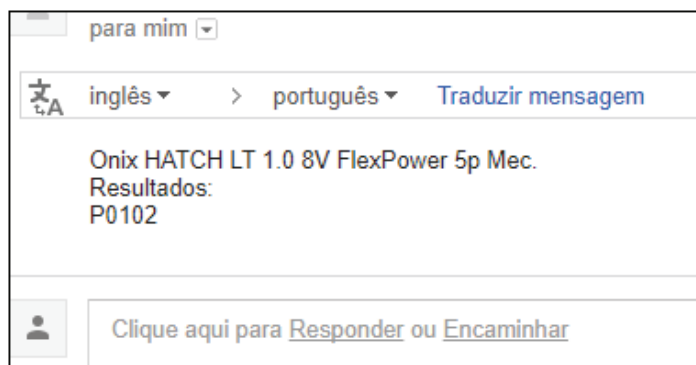
A Figura 16 apresenta a anomalia obtida, enquanto a Figura 17 apresenta o e-mail recebido.

Figura 16 – Resultados da Análise



Fonte: O autor. Coletado em 10/2017.

Figura 17 – E-mail Recebido



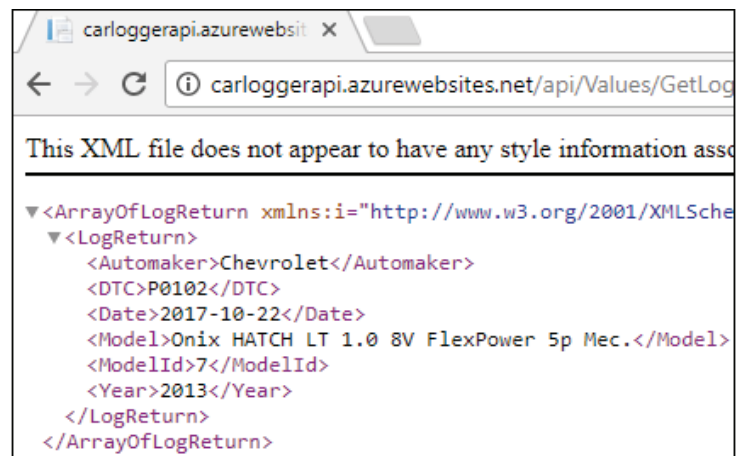
Fonte: O autor.

3) Após a identificação da anomalia pela aplicação, foi possível visualizar – por meio do método *GetLogs*, assim como por meio do banco de dados *CarLoggerDB* – que a anomalia detectada foi armazenada e estava disponível para acesso em nuvem.

Tabela 1 – Código de Anomalia Armazenado no *CarLoggerDB*

Description	ServiceDate	Code	Model_Id	Year
Onix HATCH LT 1.0 8V FlexPower 5p Mec.	2017-10-22 16:56:27.370	P0102	7	2013

Fonte: Tela do *CarLoggerDB*.

Figura 18 – Log Obtido pelo *CarLoggerAPI*

Fonte: Tela do *CarLoggerAPI*.

Assim, o teste da solução proposta foi executado com êxito e a confiabilidade do sistema constatada.

4. CONSIDERAÇÕES FINAIS

Neste estudo buscou-se, inicialmente, observar as diferentes formas de utilização dos dados coletados por meio da tecnologia *On-Board Diagnostic*. Para isto, foi efetuada uma Pesquisa Bibliográfica, cuja análise dos trabalhos obtidos apontou para o desenvolvimento de softwares com o objetivo de efetuar diferentes análises dos dados obtidos pelo OBD. Tais análises geralmente possuem como foco principal o auxílio ao condutor ou proprietário, com diferentes tipos de informações, como as relacionadas à manutenção, à má utilização do veículo, etc.

Após a análise dos resultados obtidos com a Pesquisa Bibliográfica, uma lacuna foi percebida na grande área da Computação Embarcada Automotiva: em nenhum dos trabalhos científicos analisados foi explorado o envio dos dados da interface OBDII para um banco de dados, onde pudessem ser armazenados, resgatados, processados e disponibilizados para eventuais consultas, ou seja, em nenhum dos trabalhos houve o objetivo de obter e disponibilizar os dados da interface OBD em nuvem, mas, sim, as diferentes formas para analisar tais dados.

Para explorar a lacuna descoberta, idealizou-se o desenvolvimento de uma solução de softwares. Após a execução das duas tentativas descritas, obteve-se como resultado um terceiro modelo, composto por quatro módulos e um banco de dados, além da necessidade de uso de um dispositivo ELM327. Com a comunicação via Bluetooth deste conjunto de softwares com um dispositivo ELM327 conectado à interface OBDII do veículo, torna-se possível coletar os códigos de anomalias veiculares armazenados nas ECUs dos veículos, armazená-los em um banco de dados em nuvem e disponibilizá-los para consulta e uma possível tomada de decisão.

Em testes efetuados com seis diferentes veículos, dois com o protocolo ISO 14230 KWP (2000) e outros quatro com protocolo ISO 15765-4 CAN, a solução foi capaz de obter dez códigos de anomalia armazenados nas ECUs dos veículos de forma correta, interpretá-los e armazená-los em um banco de dados em nuvem. A posterior disponibilidade de acesso foi verificada por meio de consulta ao banco de dados e por meio de requisição à API responsável pela disponibilização dos dados.

Durante a execução dos testes, a solução desenvolvida apresentou-se viável para a finalidade proposta, a qual em acréscimo aos registros dos softwares desenvolvidos, cumpre com os objetivos propostos pela dissertação.

Como continuidade da pesquisa desenvolvida, tendo-se por microfoco a abrangência da solução desenvolvida, sem que se altere sua finalidade, sugere-se que novos testes e adaptações dos softwares sejam efetuados para diferentes modelos de veículos, adequando assim a solução para protocolos de comunicação não verificados.

Para um macro foco, o qual visa aperfeiçoar a solução de modo a torná-la passível de utilização para diversas finalidades, das quais envolvem o setor automotivo como um todo, inúmeras são as possibilidades de adaptação. A utilização de ferramentas de análise de dados com tecnologia cognitiva, podem fazer com que os softwares possam não apenas apresentar as anomalias capturadas, mas também avaliem a forma com que o veículo está sendo utilizado, construindo-se uma análise preditiva de falhas causadas pela má utilização dos equipamentos e, por meio de experiências armazenadas, que possam apresentar diferentes soluções para o problema.

A utilização da solução como ferramenta de *Kaizen* pelas montadoras e de comunicação entre montadora e motorista também pode ser de grande valia. Com o aperfeiçoamento da solução desenvolvida, as montadoras podem fazer análises de seus equipamentos, podendo até identificar se as falhas ocorridas em seus veículos são devido a problemas em determinado lote de produção ou se ocorrem devido a uma falha até mesmo no projeto inicial de produção, e assim fazer uso da solução como uma ferramenta de auxílio à melhoria contínua. Com a análise e a percepção da necessidade de *recall* por meio da ferramenta, as montadoras podem utilizar o software para a notificação dos proprietários por meio de canais de comunicação, como mensagem de texto, e-mail ou redes sociais.

Ainda se tratando da adaptação da solução para uso como canal de comunicação, a solução pode ser útil para o proprietário do veículo efetuar o envio de dados de anomalias às assistências técnicas, e com isto obterem um diagnóstico à distância do veículo, por meio da ferramenta e informar ao condutor quais atitudes ele deve tomar para solucionar o problema.

REFERÊNCIAS

- ABUALI, N. *Advanced Vehicular Sensing of Road Artifacts and Driver Behavior*. IEEE, 2015.
- AGUIAR, T. R. X. Tentativa e Erro: **O que isto pode nos Ensinar sobre o Conhecimento Científico**. São Paulo: Cognito, 2011.
- AMARASINGHE, M.; KOTTEGODA, S.; ARACHCHI, A. L.; MURAMUDALIGE, S.; BANDARA, H. M. N. D.; AZEEZ, A. *Cloud-Based Driver Monitoring and Vehicle Diagnostic with OBD2 Telematics*. IEEE, 2015.
- ATHAVAN, Aravindhan; RADHIKA, N. *Memory Optimized Lifetime Vehicle Data Acquisition Framework*. IEEE, 2015.
- BAEK, S.; JANG, J. *Implementation of integrated OBD-II connector with external network*. New York: Elsevier B.V., 2014.
- BEI, X.; CHEN, N.; ZHANG, S. *On the Complexity of Trial and Error*. ACM, 2013.
- BONOMI, F.; MILITO, R.; ZHU, J.; ADDEPALLI, S. *Fog Computing and Its Role in the Internet of Things*. ACM, 2012.
- BRASIL. **Resolução CONAMA nº 315**, de 29 de outubro de 2002. Dispõe sobre a nova etapa do Programa de Controle de Emissões Veiculares – PROCONVE. Brasília, 2002. Disponível em: <<http://www.mma.gov.br/port/conama/legiabre.cfm?codlegi=337>> Acesso em: 10 fev. 2018.
- _____. **Resolução CONAMA nº 354**, de 13 de dezembro de 2004. Dispõe sobre os requisitos para adoção de sistemas de diagnose de bordo – OBD nos veículos automotores leves objetivando preservar a funcionalidade dos sistemas de controle de emissão. Brasília, 2004. Disponível em: <<http://www.mma.gov.br/port/conama/legiabre.cfm?codlegi=456>> Acesso em: 12 fev. 2018.
- CHEN, L.; LI, H.; CHANG, H.; TANG, J.; LI, K. S. *An Intelligent Vehicular Telematics Platform for Vehicle Driving Safety Supporting System*. IEEE, 2015.
- CHILIN, L.; QIN, Z. *Study on Auto Collaborative After-sales Service Based on Value Flow Technology*. ICMI, 2011.
- CHOU, Y.; MO, Y.; SU, J.; CHANG, W.; CHEN, L.; TANG, J.; YU, C. *i-Car System: A LoRa-based Low Power Wide Area Networks Vehicle Diagnostic System for Driving Safety*. IEEE-ICASI, 2017.

COTTINGHAM, D. N.; DAVIES, J. J. *A vision for wireless access on the road network*. Hamburg, Germany: [in] *Proceedings of the 4th International Workshop on Intelligent Transportation (WIT 2007)*, 2007.

DAVIES, J. J.; BERESFORD, A; HOPPER, A. *Scalable, distributed, real-time map generation*. IEEE Pervasive Computing, 2006.

DAY, P.; WU, J.; POULTON, N. *Beyond real time*. ITS International, 2006.

DZHELEKARSKI, P.; ALEXIEV, D. *Reading and Interpreting Diagnostic Data from Vehicle OBDII System*. *Electronics*, 2005.

EDWARDS, S. A. *Design Languages for Embedded Systems*. Columbia University, 2003.

ELM. *ELM327 OBD to RS232 Interpreter*. *ELM Electronics*, 2016. Disponível em: <<https://www.elmelectronics.com/wp-content/uploads/2016/07/ELM327DS.pdf>> Acesso em: 20 de out. 2017

FRIEDRICH, A.; TAPPE, M.; GARMS, S.; PALOCZ-ANDRESEN M.; SCHROLL, S. *On-Board-Diagnose (OBD) und On-Board_Messung (OBM) im Kraftfahrzeug*. WLB Wasser, Luft und Boden, 1998.

GALHARDI, A. C. *New trends in Automotive Embedded System Design*. *SAE International*, 2014.

_____. Aspectos de Produtos da Computação Embarcada Automotiva. In: Antonio César Galhardi; João José Ferreira de Aguiar; Marlene Rodrigues da Silva Aguiar. (Org.).

Contribuições para as Ciências Gerenciais. v.4. 1ed. Jundiaí-SP: Edições Brasil, 2014a.

GALLASSI, T. T.; MARTINS, L. E. G. Um Estudo Exploratório sobre Sistemas Operacionais Embarcados. *Revista Técnica Fatec Americana*, v.1. p.78-105, 2014.

GRUTESER, M.; GRUNWALD, D. *Anonymous usage of location-based services through spatial and temporal cloaking*. San Francisco, CA, USA: *Proceedings of 1st ACM/USENIX International Conference on Mobile Systems, Applications and Services (MobiSys)*, 2003.

GUIMARÃES, A. de A. **Eletrônica Embarcada Automotiva**. 1ed. São Paulo: Érica, 2010.

HACK A DAY. **Foto de uma Saída OBDII**. Disponível em: <<https://hackadaycom.files.wordpress.com/2013/10/obd-ii.jpg?w=800&h=600>>. Acesso em: 14 out. 2017.

HU, X.; GIAN, N. K.; SHEN, J.; LEUNG, V. C. M.; LI, X. *Towards Mobility-as-a-Service to Promote Smart Transportation*. Crown, 2015.

HU, Y. C. PATEL, M. SABELLA, D. SPRECHER, N. YOUNG, V. *Mobile Edge Computing A key technology towards 5G*. ETSI, 2015a.

HUSNI, E.; HERTANTYO, G. B.; WICAKSONO, D. W.; HASIBUAN, F. C.; RAHAYU, A. U.; TRIAWAN, M. A. *Applied Internet of Things (IoT): Car monitoring System Using IBM BlueMix*. IEEE, 2016.

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION – ISO. **ISO 15765-4/** SAE J 2480 CAN. *Road Vehicles – Diagnostic communication over Controller Area Network (DoCAN) – Part 4: Requirements for emissions-related systems*. ISO, 2011.

_____. **ISO 14230 KWP (2000)**. *Road vehicles - Diagnostic systems - Keyword Protocol 2000 - Part 1: Physical Layer. International Standard*, 1999.

JHOU, J.; CHEN, S.; TSAY, E.; LAI, M. *The Implementation of OBD-II Vehicle Diagnosis System Integrated with Cloud Computation Technology*. IEEE, 2013.

JIE, H.; FUWU, Y.; JING, T.; PAN, W.; KAI, C. *Developing PC-Based Automobile Diagnostic System Based on OBD System*. IEEE, 2010.

KIM, W. *Cloud Computing: Today and Tomorrow*. *Journal Of Object Technology*, 2009.

LEEN, G.; HEFFERNAN, D. *Expanding automotive electronic systems*. IEEE, 2002.

MARSTON, S.; LI, Z.; BANDYOPADHYAY, S.; ZHANG, J.; GHALSASI, A. *Cloud computing- The business perspective*. New York: Elsevier B.V., 2010.

MCGLOTHLIN, J. P.; KHAN, L. *Scalable Queries For Large Datasets Using Cloud Computing: A Case Study*. ACM, 2011.

MCKINSEY & COMPANY. *Automotive Revolution – perspective towards 2030*. *Advanced Industries*, jan. de 2016.

NAVET, N.; SONG, Y.; SIMONOT-LION, F.; WILWERT, C. *Trends in Automotive Communication Systems*. IEEE, 2005

NOGUEIRA, F. C. V. **O Estabelecimento de um Programa de Pós-Venda: um Estudo de Caso no Setor de Concessionárias de Automóveis no Brasil**. Dissertação (Mestrado em Administração) Instituto COPPEAD de Administração. Universidade Federal do Rio de Janeiro. Rio de Janeiro: UFRJ, 2002.

PALOCZ-ANDRESEN, M. *Onboard Diagnostics and Measurement in the Automotive Industry, Shipbuilding, and Aircraft Construction*. *SAE International*, 2012.

PORTAL ENGIE. **Engie**: A solução para manutenção do seu carro. Disponível em: <<https://engieapp.com/pt/>>. Acesso em: 12 jan. 2018.

PORTAL EXAME. **App do criador do Waze evita que você seja enganado por mecânicos.** Disponível em: <<https://exame.abril.com.br/tecnologia/app-do-criador-do-waze-evita-que-voce-seja-enganado-por-mecanicos/>>. Acesso em: 15 jan. 2018.

PORTAL G1. **Aplicativo ‘traduz’ dados de veículo em tempo real para motoristas.** G1-Sorocaba-Jundiaí, 2014. Disponível em: <<http://g1.globo.com/sao-paulo/sorocaba-jundiai/noticia/2014/10/aplicativo-traduz-dados-de-veiculo-em-tempo-real-para-usuarios.html>> Acesso em: 10 fev. 2018.

PORTAL GOOGLE PLAY. **Apps.** Disponível em: <https://play.google.com/store/search?q=OBD&c=apps&rating=1&hl=pt_BR>. Acesso em: 11 jan. 2018.

_____. **Torque Pro (OBD2/ Carro).** Disponível em: <https://play.google.com/store/apps/details?id=org.prowl.torque&hl=pt_BR>. Acesso em: 11 jan. 2018a.

SAE. SAE J2012 *Diagnostic Trouble Code Definitions*. **SAE International**, 2007.

SEGURA, D. C. M.; STABILE, R. S.; BRUSCHI, S. M.; SOUZA, P. S. L. **Providing Computing Services through Mobile Devices in a Collaborative Way - A Fog Computing Case Study**. ACM, 2017

SHIBU, K. V. **Introduction to Embedded Systems**. Tata McGraw-Hill, 2009.

TAPPE, M. **On-Board-Diagnosis (OBD) und On-Board-Measurement (OBM)**. Berlin: Annual Review, Federal Environmental Agency, 1997.

UNIÃO EUROPEIA. Parlamento Europeu e do Conselho. **Diretiva 98/69/CE**, de 13 de outubro de 1998. Disponível em: < <https://sogi8.sogi.com.br/Arquivo/Modulo113.MRID109/Registro7235/documento%201.pdf>>. Acesso em: 17 fev. 2018.

WATZENIG, D.; ARMENGAUD, E. *Automotive Embedded Systems*. **Elektrotechnik & Informationstechnik**, 2011.

WEAVER, S.; PETTY, L.E. *Reproducibility and Accuracy Emission Measurements Using the RAVEM™ System*. **SAE International**, 2004.

WHAIDUZZAMAN, M.; SOOKHAK, M.; GANI, A.; BUYYA, R. **A survey on vehicular cloud computing**. New York: Elsevier B.V., 2013.

WIEGAND, B. **SAE J1850 Class B Data Communications Network Interface**. IEE, 1998.

XIONG, J.; GU, H. **A Remote Engine Health Management System based on Mobile cloud computing**. IEEE, 2016.

YAN, W. **A two-year Survey on Security Challenges in Automotive Threat Landscape**. IEEE, 2015.

ZHANG, M.; CHEN, C.; WO, T.; XIE, T.; BHUIYAN, M. Z. A.; LIN, X. **SafeDrive: Online Driving Anomaly Detection From Large-Scale Vehicle Data**. IEEE, 2016.

ANEXO A - Registro do Software CarLogger



REPÚBLICA FEDERATIVA DO BRASIL
Ministério Da Indústria, Comércio Exterior e Serviços
Instituto Nacional da Propriedade Industrial

Diretoria de Patentes, Programas de Computador e Topografias de Circuitos Integrados

Certificado de Registro de Programas de Computador

Processo nº: BR 51 2018 000599-3

ANEXO B - Registro do Software CarLoggerAPI



REPÚBLICA FEDERATIVA DO BRASIL
Ministério Da Indústria, Comércio Exterior e Serviços
Instituto Nacional da Propriedade Industrial

Diretoria de Patentes, Programas de Computador e Topografias de Circuitos Integrados

Certificado de Registro de Programas de Computador

Processo nº: BR 51 2018 000600-0

ANEXO C - Registro do Software CarLoggerDB

REPÚBLICA FEDERATIVA DO BRASIL
Ministério Da Indústria, Comércio Exterior e Serviços
Instituto Nacional da Propriedade Industrial

Diretoria de Patentes, Programas de Computador e Topografias de Circuitos Integrados

Certificado de Registro de Programas de Computador

Processo nº: **BR 51 2018 000601-9**

ANEXO D - Registro do Software DataModel

REPÚBLICA FEDERATIVA DO BRASIL
Ministério Da Indústria, Comércio Exterior e Serviços
Instituto Nacional da Propriedade Industrial

Diretoria de Patentes, Programas de Computador e Topografias de Circuitos Integrados

Certificado de Registro de Programas de Computador

Processo nº: **BR 51 2018 000602-7**