

TURNO:	<b>NOTURNO</b>	VERSÃO:	2	ANO / SEMESTRE:	2016.1	Nº	
--------	----------------	---------	---	-----------------	--------	----	--

UNIVERSIDADE REGIONAL DE BLUMENAU  
CENTRO DE CIÊNCIAS EXATAS E NATURAIS  
DEPARTAMENTO DE SISTEMAS E COMPUTAÇÃO  
CURSO DE CIÊNCIA DA COMPUTAÇÃO — BACHARELADO  
COORDENAÇÃO DE TRABALHO DE CONCLUSÃO DE CURSO

## PROPOSTA PARA O TRABALHO DE CONCLUSÃO DE CURSO

**TÍTULO:** OBD-JRP: MONITORAMENTO VEICULAR COM JAVA E RASPBERRY PI

**ÁREA:** Internet das coisas

**Palavras-chave:** Monitoramento veicular. OBD. Raspberry Pi. Java. Bluetooth.

## 1 IDENTIFICAÇÃO

### 1.1 ALUNO

Nome: Ricardo Artur Staroski		Código/matricula: 81281/23636	
Endereço residencial:			
Rua: Ricardo Georg		nº: 1632	Complemento: Casa
Bairro: Itoupava Central	CEP: 89069-101	Cidade: Blumenau	UF: SC
Telefone fixo: (47) 3337-4969		Celular: (47) 8826-3399	
Endereço comercial:			
Empresa: Senior Sistemas			
Rua: São Paulo		nº: 825	Bairro: Victor Konder
CEP: 89012-001	Cidade: Blumenau	UF: SC	Telefone: (47) 3221-3300
E-Mail FURB: staroski@inf.furb.br		E-Mail alternativo: ricardo.staroski@gmail.com	

### 1.2 ORIENTADOR

Nome: Miguel Alexandre Wisintainer	
E-Mail FURB: maw@furb.br	E-Mail alternativo: tcpipchip@hotmail.com

## **2 DECLARAÇÕES**

### **2.1 DECLARAÇÃO DO ALUNO**

Declaro que estou ciente do Regulamento do Trabalho de Conclusão de Curso de Ciência da Computação e que a proposta em anexo, a qual concordo, foi por mim rubricada em todas as páginas. Ainda me comprometo pela obtenção de quaisquer recursos necessários para o desenvolvimento do trabalho, caso esses recursos não sejam disponibilizados pela Universidade Regional de Blumenau (FURB).

Assinatura: \_\_\_\_\_ Local/data: \_\_\_\_\_

### **2.2 DECLARAÇÃO DO ORIENTADOR**

Declaro que estou ciente do Regulamento do Trabalho de Conclusão do Curso de Ciência da Computação e que a proposta em anexo, a qual concordo, foi por mim rubricada em todas as páginas. Ainda me comprometo a orientar o aluno da melhor forma possível de acordo com o plano de trabalho explícito nessa proposta.

Assinatura: \_\_\_\_\_ Local/data: \_\_\_\_\_

### 3 AVALIAÇÃO DA PROPOSTA

#### 3.1 AVALIAÇÃO DO(A) ORIENTADOR(A)

Acadêmico(a): Ricardo Artur Staroski

Orientador(a): Miguel Alexandre Wisintainer

ASPECTOS AVALIADOS		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO 1.1. O tema de pesquisa está devidamente contextualizado/delimitado?			
	1.2. O problema está claramente formulado?			
	2. OBJETIVOS 2.1. O objetivo geral está claramente definido e é passível de ser alcançado?			
	2.2. São apresentados objetivos específicos (opcionais) coerentes com o objetivo geral? Caso não sejam apresentados objetivos específicos, deixe esse item em branco.			
	3. RELEVÂNCIA 3.1. A proposta apresenta um grau de relevância em computação que justifique o desenvolvimento do TCC?			
	4. METODOLOGIA 4.1. Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	4.2. Os métodos e recursos estão devidamente descritos e são compatíveis com a metodologia proposta?			
	4.3. A proposta apresenta um cronograma físico (período de realização das etapas) de maneira a permitir a execução do TCC no prazo disponível?			
	5. REVISÃO BIBLIOGRÁFICA 5.1. As informações apresentadas são suficientes e têm relação com o tema do TCC?			
	5.2. São apresentados trabalhos correlatos, bem como comentadas as principais características dos mesmos?			
	6. REQUISITOS DO SISTEMA A SER DESENVOLVIDO 6.1. Os requisitos funcionais e não funcionais do sistema a ser desenvolvido foram claramente descritos?			
	7. CONSIDERAÇÕES FINAIS 7.1. As considerações finais relacionam os assuntos apresentados na revisão bibliográfica com a realização do TCC?			
ASPECTOS METODOLÓGICOS	8. REFERÊNCIAS BIBLIOGRÁFICAS 8.1. As referências bibliográficas obedecem às normas da ABNT?			
	8.2. As referências bibliográficas contemplam adequadamente os assuntos abordados na proposta (são usadas obras atualizadas e/ou as mais importantes da área)?			
	9. CITAÇÕES 9.1. As citações obedecem às normas da ABNT?			
	9.2. As informações retiradas de outros autores estão devidamente citadas?			
	10. AVALIAÇÃO GERAL (organização e apresentação gráfica, linguagem usada) 10.1. O texto obedece ao formato estabelecido?			
	10.2. A exposição do assunto é ordenada (as idéias estão bem encadeadas e a linguagem utilizada é clara)?			
<p>A proposta de TCC deverá ser revisada, isto é, necessita de complementação, se:</p> <ul style="list-style-type: none"> <li>qualquer um dos itens tiver resposta NÃO ATENDE;</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS TÉCNICOS</b> tiverem resposta ATENDE PARCIALMENTE; ou</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS METODOLÓGICOS</b> tiverem resposta ATENDE PARCIALMENTE.</li> </ul> <p><b>PARECER:</b> ( ) APROVADA ( ) NECESSITA DE COMPLEMENTAÇÃO</p>				

Assinatura do(a) avaliador(a): \_\_\_\_\_

Local/data: \_\_\_\_\_

CONSIDERAÇÕES DO(A) ORIENTADOR(A):

Caso o(a) orientador(a) tenha assinalado em sua avaliação algum item como “atende parcialmente”, devem ser relatos os problemas/melhorias a serem efetuadas.

Na segunda versão, caso as alterações sugeridas pelos avaliadores não sejam efetuadas, deve-se incluir uma justificativa.

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

Assinatura do(a) avaliador(a):

Local/data:

### 3.2 AVALIAÇÃO/HOMOLOGAÇÃO DO COORDENADOR DE TCC

Acadêmico(a): Ricardo Artur Staroski

Avaliador(a): Maurício Capobianco Lopes

ASPECTOS AVALIADOS		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO 1.1. O tema de pesquisa está devidamente contextualizado/delimitado?			
	1.2. O problema está claramente formulado?			
	2. OBJETIVOS 2.1. O objetivo geral está claramente definido e é passível de ser alcançado?			
	2.2. São apresentados objetivos específicos (opcionais) coerentes com o objetivo geral? Caso não sejam apresentados objetivos específicos, deixe esse item em branco.			
	3. RELEVÂNCIA 3.1. A proposta apresenta um grau de relevância em computação que justifique o desenvolvimento do TCC?			
	4. METODOLOGIA 4.1. Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	4.2. Os métodos e recursos estão devidamente descritos e são compatíveis com a metodologia proposta?			
	4.3. A proposta apresenta um cronograma físico (período de realização das etapas) de maneira a permitir a execução do TCC no prazo disponível?			
	5. REVISÃO BIBLIOGRÁFICA 5.1. As informações apresentadas são suficientes e têm relação com o tema do TCC?			
	5.2. São apresentados trabalhos correlatos, bem como comentadas as principais características dos mesmos?			
	6. REQUISITOS DO SISTEMA A SER DESENVOLVIDO 6.1. Os requisitos funcionais e não funcionais do sistema a ser desenvolvido foram claramente descritos?			
	7. CONSIDERAÇÕES FINAIS 7.1. As considerações finais relacionam os assuntos apresentados na revisão bibliográfica com a realização do TCC?			
ASPECTOS METODOLÓGICOS	8. REFERÊNCIAS BIBLIOGRÁFICAS 8.1. As referências bibliográficas obedecem às normas da ABNT?			
	8.2. As referências bibliográficas contemplam adequadamente os assuntos abordados na proposta (são usadas obras atualizadas e/ou as mais importantes da área)?			
	9. CITAÇÕES 9.1. As citações obedecem às normas da ABNT?			
	9.2. As informações retiradas de outros autores estão devidamente citadas?			
	10. AVALIAÇÃO GERAL (organização e apresentação gráfica, linguagem usada) 10.1. O texto obedece ao formato estabelecido?			
	10.2. A exposição do assunto é ordenada (as idéias estão bem encadeadas e a linguagem utilizada é clara)?			
<p>A proposta de TCC deverá ser revisada, isto é, necessita de complementação, se:</p> <ul style="list-style-type: none"> <li>qualquer um dos itens tiver resposta NÃO ATENDE;</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS TÉCNICOS</b> tiverem resposta ATENDE PARCIALMENTE; ou</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS METODOLÓGICOS</b> tiverem resposta ATENDE PARCIALMENTE.</li> </ul> <p><b>PARECER:</b> ( ) APROVADA ( ) NECESSITA DE COMPLEMENTAÇÃO</p>				

OBSERVAÇÕES:

Assinatura do(a) avaliador(a):

Local/data:

### 3.3 AVALIAÇÃO DO PROFESSOR DA DISCIPLINA DE TCCI

Acadêmico(a): Ricardo Artur Staroski

Avaliador(a): Roberto Heinzle

ASPECTOS AVALIADOS		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO 1.1. O tema de pesquisa está devidamente contextualizado/delimitado?			
	1.2. O problema está claramente formulado?			
	2. OBJETIVOS 2.1. O objetivo geral está claramente definido e é passível de ser alcançado?			
	2.2. São apresentados objetivos específicos (opcionais) coerentes com o objetivo geral? Caso não sejam apresentados objetivos específicos, deixe esse item em branco.			
	3. RELEVÂNCIA 3.1. A proposta apresenta um grau de relevância em computação que justifique o desenvolvimento do TCC?			
	4. METODOLOGIA 4.1. Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	4.2. Os métodos e recursos estão devidamente descritos e são compatíveis com a metodologia proposta?			
	4.3. A proposta apresenta um cronograma físico (período de realização das etapas) de maneira a permitir a execução do TCC no prazo disponível?			
	5. REVISÃO BIBLIOGRÁFICA 5.1. As informações apresentadas são suficientes e têm relação com o tema do TCC?			
	5.2. São apresentados trabalhos correlatos, bem como comentadas as principais características dos mesmos?			
	6. REQUISITOS DO SISTEMA A SER DESENVOLVIDO 6.1. Os requisitos funcionais e não funcionais do sistema a ser desenvolvido foram claramente descritos?			
	7. CONSIDERAÇÕES FINAIS 7.1. As considerações finais relacionam os assuntos apresentados na revisão bibliográfica com a realização do TCC?			
ASPECTOS METODOLÓGICOS	8. REFERÊNCIAS BIBLIOGRÁFICAS 8.1. As referências bibliográficas obedecem às normas da ABNT?			
	8.2. As referências bibliográficas contemplam adequadamente os assuntos abordados na proposta (são usadas obras atualizadas e/ou as mais importantes da área)?			
	9. CITAÇÕES 9.1. As citações obedecem às normas da ABNT?			
	9.2. As informações retiradas de outros autores estão devidamente citadas?			
	10. AVALIAÇÃO GERAL (organização e apresentação gráfica, linguagem usada) 10.1. O texto obedece ao formato estabelecido?			
	10.2. A exposição do assunto é ordenada (as idéias estão bem encadeadas e a linguagem utilizada é clara)?			
PONTUALIDADE NA ENTREGA			atraso de _____ dias	
<p>A proposta de TCC deverá ser revisada, isto é, necessita de complementação, se:</p> <ul style="list-style-type: none"> <li>qualquer um dos itens tiver resposta NÃO ATENDE;</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS TÉCNICOS</b> tiverem resposta ATENDE PARCIALMENTE; ou</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS METODOLÓGICOS</b> tiverem resposta ATENDE PARCIALMENTE.</li> </ul> <p><b>PARECER:</b> ( ) APROVADA ( ) NECESSITA DE COMPLEMENTAÇÃO</p>				

OBSERVAÇÕES:

Assinatura do(a) avaliador(a): \_\_\_\_\_

Local/data: \_\_\_\_\_

### 3.4 AVALIAÇÃO DO(A) PROFESSOR(A) ESPECIALISTA NA ÁREA

Acadêmico(a): Ricardo Artur Staroski

Avaliador(a): Luciana Pereira de Araújo

ASPECTOS AVALIADOS		atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO 1.1. O tema de pesquisa está devidamente contextualizado/delimitado?			
	1.2. O problema está claramente formulado?			
	2. OBJETIVOS 2.1. O objetivo geral está claramente definido e é passível de ser alcançado?			
	2.2. São apresentados objetivos específicos (opcionais) coerentes com o objetivo geral? Caso não sejam apresentados objetivos específicos, deixe esse item em branco.			
	3. RELEVÂNCIA 3.1. A proposta apresenta um grau de relevância em computação que justifique o desenvolvimento do TCC?			
	4. METODOLOGIA 4.1. Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?			
	4.2. Os métodos e recursos estão devidamente descritos e são compatíveis com a metodologia proposta?			
	4.3. A proposta apresenta um cronograma físico (período de realização das etapas) de maneira a permitir a execução do TCC no prazo disponível?			
	5. REVISÃO BIBLIOGRÁFICA 5.1. As informações apresentadas são suficientes e têm relação com o tema do TCC?			
	5.2. São apresentados trabalhos correlatos, bem como comentadas as principais características dos mesmos?			
	6. REQUISITOS DO SISTEMA A SER DESENVOLVIDO 6.1. Os requisitos funcionais e não funcionais do sistema a ser desenvolvido foram claramente descritos?			
	7. CONSIDERAÇÕES FINAIS 7.1. As considerações finais relacionam os assuntos apresentados na revisão bibliográfica com a realização do TCC?			
ASPECTOS METODOLÓGICOS	8. REFERÊNCIAS BIBLIOGRÁFICAS 8.1. As referências bibliográficas obedecem às normas da ABNT?			
	8.2. As referências bibliográficas contemplam adequadamente os assuntos abordados na proposta (são usadas obras atualizadas e/ou as mais importantes da área)?			
	9. CITAÇÕES 9.1. As citações obedecem às normas da ABNT?			
	9.2. As informações retiradas de outros autores estão devidamente citadas?			
	10. AVALIAÇÃO GERAL (organização e apresentação gráfica, linguagem usada) 10.1. O texto obedece ao formato estabelecido?			
	10.2. A exposição do assunto é ordenada (as idéias estão bem encadeadas e a linguagem utilizada é clara)?			
<p>A proposta de TCC deverá ser revisada, isto é, necessita de complementação, se:</p> <ul style="list-style-type: none"> <li>qualquer um dos itens tiver resposta NÃO ATENDE;</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS TÉCNICOS</b> tiverem resposta ATENDE PARCIALMENTE; ou</li> <li>pelo menos <b>4 (quatro)</b> itens dos <b>ASPECTOS METODOLÓGICOS</b> tiverem resposta ATENDE PARCIALMENTE.</li> </ul> <p><b>PARECER:</b> ( ) APROVADA ( ) NECESSITA DE COMPLEMENTAÇÃO</p>				

OBSERVAÇÕES:

Assinatura do(a) avaliador(a):

Local/data:

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE CIÊNCIA DA COMPUTAÇÃO – BACHARELADO**

**OBD-JRP: MONITORAMENTO VEICULAR COM JAVA E**  
**RASPBERRY PI**

**RICARDO ARTUR STAROSKI**

**BLUMENAU**  
**2016**



**RICARDO ARTUR STAROSKI**

**OBD-JRP: MONITORAMENTO VEICULAR COM JAVA E  
RASPBERRY PI**

Proposta de Trabalho de Conclusão de Curso  
submetida à Universidade Regional de  
Blumenau para a obtenção dos créditos na  
disciplina Trabalho de Conclusão de Curso I  
do curso de Ciência da Computação —  
Bacharelado.

Prof. Miguel Alexandre Wisintainer - Orientador

**BLUMENAU  
2016**

## 1 INTRODUÇÃO

A Internet das Coisas ou IOT, *Internet of Things*, está presente no cotidiano e todos os dias novos participantes participam dessa evolução. Segundo Ng (2015), estes participantes vão desde dispositivos ou aparelhos contribuindo para a automação residencial até roupas inteligentes. O número de “coisas” gerando volumes de dados de forma automática e consistente aumenta diariamente (Ng, 2015).

O que aconteceu? Em meados da década de 80, a *General Motors* tomou medidas analógicas e parâmetros de entrada dos motores, em seguida usou uma tabela de pesquisa em um *chip* de memória para produzir saídas pré-computadas (Ng, 2015). Com a redução de preço dos componentes eletrônicos e maior sofisticação dos mesmos, abriu-se caminho para a inovação. Segundo Ng (2015), atualmente motores industriais possuem centenas de sensores que gravam milhares de medições. O principal uso dessas medições é auxiliar a identificar a causa raiz de falhas de forma a facilitar o processo de reparação.

O que está acontecendo agora? Grande parte dos dispositivos domésticos incluem conectividade *WiFi* ou *Bluetooth* permitindo a comunicação com outros dispositivos e aparelhos (Ng, 2015). Segundo Ng (2015), a capacidade de aplicar modelos analíticos em tempo real mudou para sempre a Internet das Coisas, permitindo que algumas empresas implementassem sistemas preditivos e analíticos de forma eficiente.

No meio automobilístico, a *Resolução CONAMA n° 354* (2004) considera que Sistemas de Diagnose de Bordo – OBD, constituem tecnologia de ação comprovada na identificação de mau funcionamento do veículo. Possibilitando, através da análise dos dados, prevenir a ocorrência de avarias aos sistemas do veículo, contribuindo para a qualidade ambiental (*Resolução...*, 2004).

Diante do exposto, este trabalho propõe o desenvolvimento de um protótipo de *software* embarcado para coletar e disponibilizar as informações do Sistema de Diagnose de Bordo – OBD de um carro.

### 1.1 SISTEMAS DE DIAGNOSE DE BORDO – OBD

A especificação de um sistema capaz de recolher informações e estabelecer os diagnósticos de bordo é vantajosa para o dono do veículo, bem como para um técnico de reparação (ZURAWSKI, 2009, p. 33, tradução nossa). O termo utilizado para esta função é "diagnose de bordo" ou OBD, *On Board Diagnostic*. O conceito OBD refere-se ao auto diagnóstico e reporte do estado dos componentes do carro, Zurawski (2009) complementa que o OBD só se tornou possível devido à introdução de sistemas computadorizados nos veículos.

O papel das funções de diagnóstico predecessoras ao OBD era limitado a piscar uma luz assim que um problema específico fosse detectado. Zurawski (2009) explica que os sistemas OBD recentes são baseados na padronização da comunicação, dos dados monitorados e dos códigos de uma lista de falhas específicas.

## 1.2 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é o desenvolvimento de um *software* embarcado, para ler os dados da porta OBD2 de um carro, enviar estes dados para um servidor.

Os objetivos específicos do trabalho são:

- a) desenvolver o *firmware*, que irá monitorar a porta OBD2 do carro e os enviar os dados lidos para um servidor;
- b) desenvolver o *software* servidor, que irá receber os dados enviados pelo *firmware* e armazenar os mesmos;
- c) desenvolver uma página *web* para consultar o histórico de leituras.

## 1.3 RELEVÂNCIA DO TRABALHO

O trabalho proposto é de relevância acadêmica, pois a especificação OBD2 tem sido pouco explorada no meio, inclusive não há registro de trabalhos sobre OBD2 nos cursos de Ciência da Computação e Sistemas de Informação da FURB. No campo profissional o trabalho é relevante por expor os aspectos da especificação OBD2 e os parâmetros do veículo que podem ser monitorados, podendo servir de base para o desenvolvimento de soluções comerciais. No aspecto social a relevância se dá pelo fato de que, segundo *Resolução...* (2004), relatórios com informações sobre os parâmetros dos veículos podem permitir a prevenção de danos aos sistemas de controle de emissão de gases.

## 1.4 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

- a) levantamento bibliográfico: pesquisar e relacionar informações sobre os assuntos abordados no trabalho;
- b) definição de requisitos: definir os requisitos do protótipo de acordo com as necessidades encontradas no levantamento bibliográfico, no estudo dos trabalhos correlatos e nas reuniões com o orientador;
- c) elicitação de requisitos: reavaliar os requisitos de acordo com os objetivos do trabalho e levantamento bibliográfico;

- d) especificação do protótipo: utilizar a ferramenta Papyrus para elaborar os diagramas de componentes, de casos de uso e de classes, tanto do *firmware* quanto do servidor, de forma que se possa ter uma visão geral da arquitetura e desenvolvimento do protótipo;
- e) desenvolvimento do *firmware*: desenvolver o *software* embarcado que irá monitorar a porta OBD2 do carro e transmitir os dados lidos para o *software* servidor;
- f) desenvolvimento do servidor: desenvolver o *software* servidor que irá receber as informações enviadas pelo *firmware* e armazenar as mesmas;
- g) desenvolvimento da página *web*: desenvolver a página *web* para consultar o histórico das leituras persistidas no servidor;
- h) teste de campo: verificar a utilização real em um carro que atenda à especificação OBD2, de modo a confirmar o funcionamento do protótipo.

As etapas serão realizadas nos períodos relacionados no Quadro 1.

### Quadro 1 - Cronograma

[illegible]

## 2 REVISÃO BIBLIOGRÁFICA

Este capítulo tem como objetivo explorar os principais assuntos necessários para a realização deste trabalho. Os assuntos foram subdivididos em cinco partes, onde a seção 2.1 apresenta a origem do OBD. A seção 2.2 expõe detalhes técnicos do OBD2. A seção 2.3 apresenta a interface ELM327. A seção 2.4 apresenta a plataforma *Raspberry Pi* e, por fim, na seção 2.5 são descritos dois trabalhos correlatos.

### 2.1 HISTÓRIA DO OBD

A denominação OBD deriva do inglês "*On Board Diagnostic*" e significa "Diagnose de Bordo". Este diagnóstico é realizado pelas próprias unidades eletrônicas do veículo. Segundo Manavella (2009), em 1988 o CARB, Comitê de Administração dos Recursos do Ar da Califórnia, estabeleceu uma norma não padronizada denominada OBD1 para que todos os veículos vendidos no estado da Califórnia, nos EUA, incorporassem em sua unidade de comando um sistema de diagnóstico capaz de detectar defeitos nos elementos e sistemas de controle de emissões. Manavella (2009) complementa ainda que o OBD1 especificava um indicador luminoso chamado MIL, *Malfunction Indicator Lamp*, que acendia na presença de falhas. O indicador MIL, em português é chamado de LIM, Lâmpada Indicadora de Mau funcionamento (Resolução..., 2004).

#### 2.1.1 OBD NO BRASIL

No Brasil, o Conselho Nacional do Meio Ambiente - CONAMA determinou a introdução dos sistemas de diagnose de bordo, em duas etapas complementares e consecutivas denominadas OBDBr-1 e OBDBr-2. De acordo com *Resolução...* (2004), o sistema OBDBr-1 foi implantado em sua totalidade em 1º de janeiro de 2009 e definiu as características mínimas para a detecção de falhas nos seguintes componentes, quando aplicável:

- a) sensor de pressão absoluta ou fluxo de ar;
- b) sensor de posição da borboleta;
- c) sensor de temperatura de arrefecimento;
- d) sensor de temperatura de ar;
- e) sensor de oxigênio;
- 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;

- j) sensor para detecção de detonação;
- k) válvulas injetoras;
- l) sistema de ignição;
- m) módulo controle eletrônico do motor;
- n) lâmpada indicadora de mau funcionamento; e
- o) outros componentes que o fabricante julgue relevantes para a correta avaliação do funcionamento do veículo e controle de emissões de poluentes.

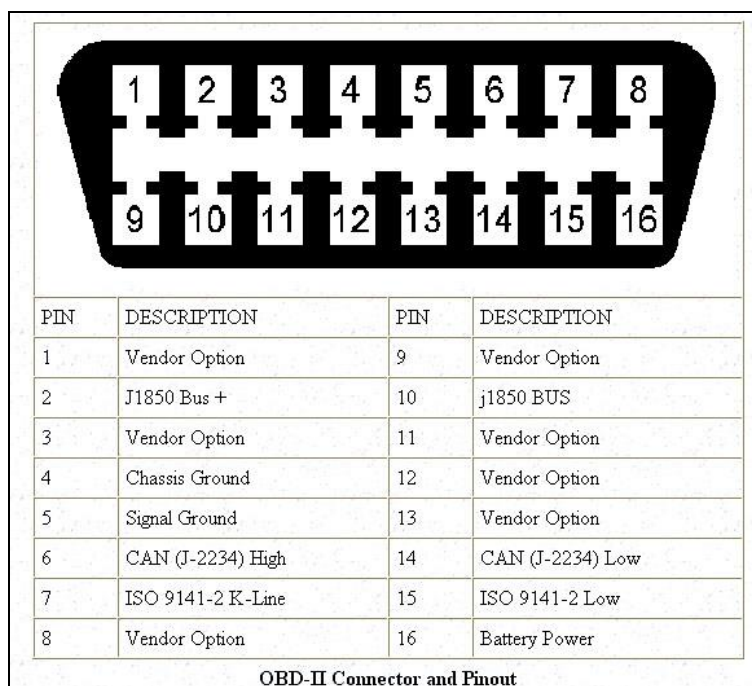
De acordo com *Resolução...* (2004), o sistema OBDBr-2 foi implantado em totalidade em 1º de janeiro de 2011 complementando as funções e características do sistema OBDBr-1, devendo detectar e registrar a existência de falhas, deterioração dos sensores de oxigênio e eficiência de conversão do catalisador, bem como apresentar características mínimas para a detecção de falhas nos seguintes componentes, quando aplicável:

- a) sensores de oxigênio (pré e pós-catalisador);
- b) eletroválvula do cânister; e
- c) outros componentes que o fabricante julgue relevantes para a correta avaliação do funcionamento do veículo e controle de emissões de poluentes.

## 2.2 OBD2

Não demorou muito para o CARB concluir que o padrão OBD1 não era eficiente para determinar o elemento que provocara o defeito. Portanto o CARB desenvolveu um novo conjunto de especificações, surgindo assim a norma OBD2 (MANAVELLA, 2009, p. 121). A SAE International (2002), “*Society of Automotive Engineers*” estabeleceu a norma SAE J1962, que determinou o conector J1962 fêmea de 16 pinos, como a interface de hardware padrão para o OBD2. Na Figura 1 observa-se o aspecto e pinagem do conector J1962.

Figura 1 – Conector J1962 e respectiva pinagem



Fonte: RIORAND (2016)

Além do conector físico, SAE International (2002) também estabeleceu a norma SAE J1979, que define o método de requisição de dados de diagnóstico e uma lista dos parâmetros padrões disponíveis na ECU, “*Electronic Control Unit*”. Os diversos parâmetros que podem ser consultados são identificados como PID’s, “*Parameter Identification Numbers*” e os códigos de erros são identificados como DTC’s, “*Diagnostic Trouble Codes*”. SAE International (2002) complementa que não exige que os fabricantes implementem todos os PID’s e permite a inclusão de PID’s proprietários, não listados na norma. SAE International (2002) também determina que o sistema de requisição e consulta permita o acesso em tempo real aos PID’s e DTC’s do veículo.

### 2.2.1 PROTOCOLOS OBD2

Enquanto a porta OBD2 é normalizada em todo o mundo, vários protocolos de comunicação continuam possíveis, dependendo dos fabricantes de veículos (TOTALCAR, 2016, tradução nossa). Atualmente estes protocolos podem ser classificados em três famílias: Redes CAN, Linhas K/L e SAE J1850.

#### 2.2.1.1 REDES CAN

Segundo TOTALCAR (2016), redes CAN utilizam os pinos 6 e 14 do conector J1962 e compreendem os seguintes protocolos:

- a) ISO 15765: Utilizado por todos os veículos. Velocidade de comunicação de 125 a

500 Kbps;

- b) SAE J1939: Utilizado principalmente por veículos pesados como caminhões e máquina agrícolas. Velocidade de comunicação de 125 a 500 Kbps.

#### 2.2.1.2 LINHAS K/L

Segundo TOTALCAR (2016), Linhas K/L utilizam os pinos 7 e 15 do conector J1962 e compreendem os seguintes protocolos:

- a) ISO 9141-2: Utilizado principalmente por fabricantes europeus. Velocidade de comunicação de 10,4 Kbps;
- b) ISO 14230 (KWP2000): Utilizado principalmente por fabricantes europeus. Dentro deste protocolo, existem dois sub protocolos que diferem no tempo de inicialização.

*Slow init*, “inicialização lenta” com velocidade de comunicação de 1,4 a 10,4 Kbps.

*Fast init*, “inicialização rápida” com velocidade fixa de 10,4 Kbps.

#### 2.2.1.3 SAE J1850

Segundo TOTALCAR (2016), SAE J1850 compreende os seguintes protocolos:

- a) PWM: utilizado principalmente pela *Ford Motors*. Velocidade de comunicação de 41,6 Kbps. Utiliza os pinos 2 e 10 do conector J1962;
- b) VPW: utilizado principalmente pela *General Motors*. Velocidade de comunicação de 10,4 a 41,6 Kbps. Utiliza somente o pino 2 do conector J1962.

### 2.2.2 MODOS DE DIAGNÓSTICO

Independente do protocolo utilizado, o padrão OBD2 define 10 modos de diagnóstico. Não necessariamente todos os modos são suportados pela ECU. Quanto mais recente for o veículo, maior é a chance de haver suporte a mais modos (OUTILS OBD FACILE, 2016).

- a) modo 1: Retorna valores comuns de alguns sensores como por exemplo, rotações do motor, velocidade do veículo, temperatura do motor, sensores de oxigênio e mistura ar/combustível. Cada sensor é identificado por um número chamado PID, utilizado para identificar o parâmetro;
- b) modo 2: Obtém um “instantâneo” de uma falha. Quando a ECU detecta uma falha, ela grava os dados do sensor naquele momento específico;
- c) modo 3: Apresenta os DTC’s armazenados. Segundo Outils OBD Facile (2016), estes códigos são padrão para todas as marcas de veículos e são divididos em quatro categorias:



- a. P0xxx: Para falhas associadas ao motor e transmissão;
  - b. C0xxx: Para falhas associadas ao chassi;
  - c. B0xxx: para falhas associadas à carroceria;
  - d. U0xxx: para falhas associadas à comunicação de rede.
- d) modo 4: Utilizado para apagar os DTC's gravados e desligar o MIL;
  - e) modo 5: Retorna o autodiagnostico do sensor lambda. Segundo Outils OBD Facile (2016), este modo não é mais utilizado e o modo 6 substitui suas funções;
  - f) modo 6: Retorna os resultados do autodiagnostico realizado nos diversos sistemas do veículo;
  - g) modo 7: Este modo retorna DTC's não confirmados. Segundo Outils OBD Facile (2016), é bastante útil após um reparo no veículo, para confirmar que um DTC não está mais presente. Seus códigos são idênticos aos do modo 3;
  - h) modo 8: Segundo The Best OBD2 Scanners (2016), diferente dos outros modos que servem somente para ler informações, este modo permite ler e gravar informações;
  - i) modo 9: Este modo obtém informações do veículo como por exemplo seu número de identificação;
  - j) modo 10: Este modo obtém os DTC's permanentes que, diferente dos modos 3 e 7, não podem ser apagados utilizando o modo 4. Outils OBD Facile (2016) explica que estes DTC's são apagados automaticamente após rodar vários quilômetros sem que se repitam.

### 2.3 INTERFACE ELM327

Segundo TOTALCAR (2016), existem vários tipos de interface OBD2 e as mais comuns utilizam o circuito ELM327, de acordo com ELM (2016), ele suporta todos os protocolos OBD2. TOTALCAR (2016) explica que existem 4 tipos de interface ELM327:

- a) ELM327 RS232: Conexão serial que está gradativamente desaparecendo nos computadores modernos. A Figura 2 apresenta o aspecto desta interface;

Figura 2 – Interface ELM327 RS232



Fonte: TOTALCAR (2016)

- b) ELM327 USB: Conexão USB, *Universal Serial Bus*, presente na maioria dos computadores atuais. A Figura 3 apresenta o aspecto desta interface;

Figura 3 – Interface ELM327 USB



Fonte: TOTALCAR (2016)

- c) ELM327 *Bluetooth*: Conexão sem fio, que pode ser utilizada com computadores ou *smartphones*. A Figura 4 apresenta o aspecto desta interface;

Figura 4 – Interface ELM327 *Bluetooth*



Fonte: TOTALCAR (2016)

- d) ELM327 *WiFi*: Conexão sem fio que pode ser utilizada com computadores ou *smartphones*. Seu aspecto é idêntico ao da interface *Bluetooth*, como ilustra a Figura 5.

Figura 5 – Interface ELM327 *WiFi*



Fonte: TOTALCAR (2016)

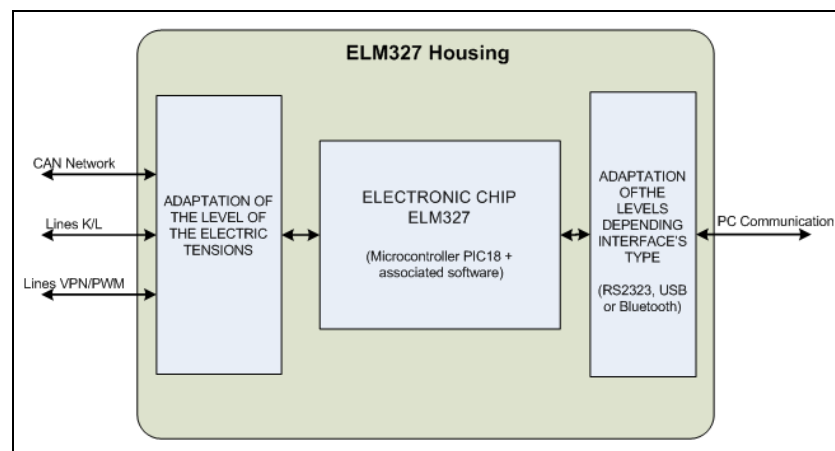
Apesar das aparências, estas 4 interfaces são eletronicamente idênticas. Somente o seu aspecto externo e o tipo de conexão são diferentes. No seu interior reside um circuito ELM327 (TOTALCAR, 2016, tradução nossa). Para operar a interface, a unidade eletrônica é composta dos seguintes blocos, representados graficamente na Figura 6:

- a) adaptadores de tensão elétrica: as redes *on-board* nos carros possuem níveis de

tensão que requerem *drivers* específicos. Como o ELM327 suporta diversos protocolos, diversos *drivers* são necessários;

- b) *chip* ELM327: é o circuito integrado, cujo nome é aplicado ao dispositivo como um todo. Ele seleciona o protocolo e o converte para um protocolo reconhecido por *modems* de computador. Ele atua como uma ponte entre os protocolos;
- c) adaptadores de tensão para o computador: o chip por si só não é hábil para se comunicar com o computador, ele precisa adaptar os níveis de tensão antes de enviar o fluxo de dados.

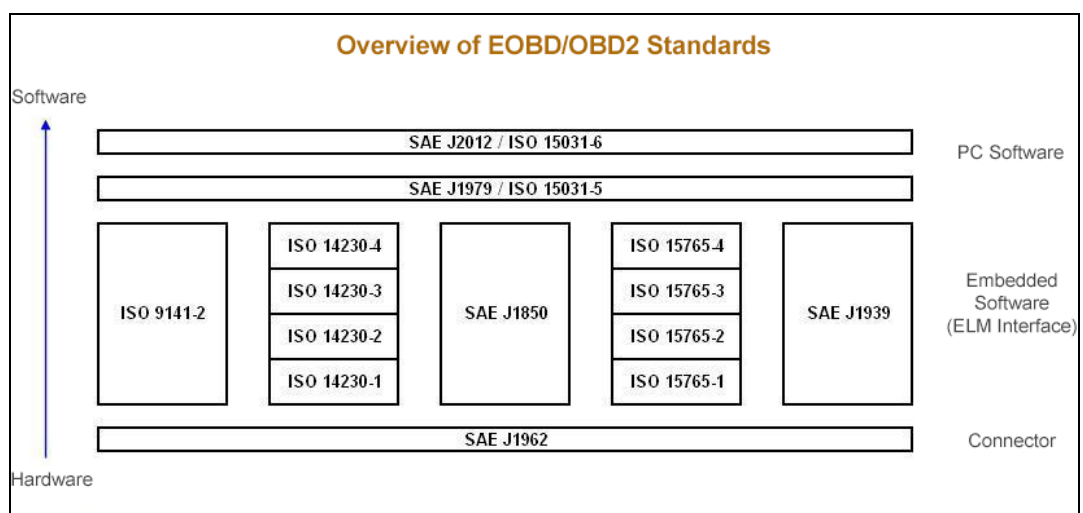
Figura 6 – Blocos da interface ELM327



Fonte: TOTALCAR (2016)

Na Figura 7 observa-se a representação em colunas dos protocolos ISO 9141-2, ISO 14230, SAE J1850, ISO 15765 e SAE J1979. Segundo TOTALCAR (2016), o papel do ELM327 é decodificar estes vários protocolos de comunicação.

Figura 7 – Visão geral dos protocolos de comunicação

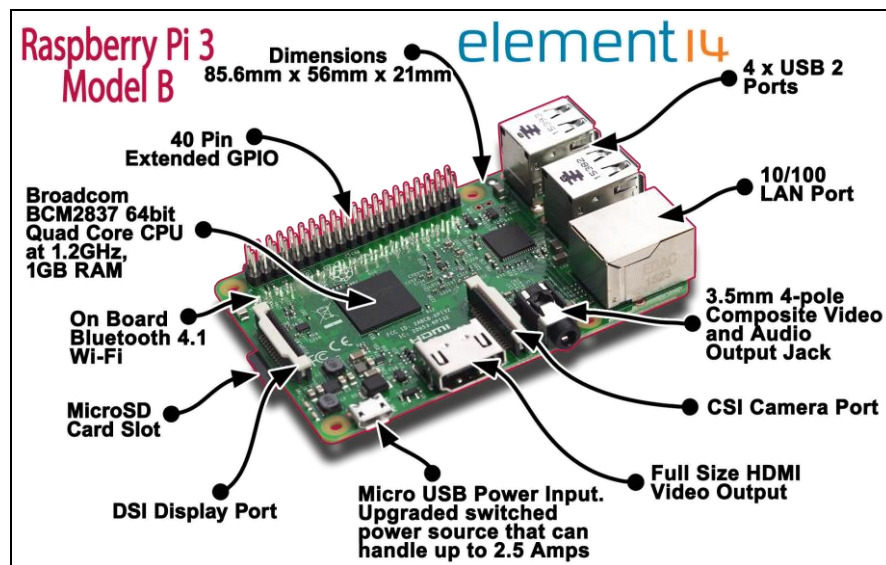


Fonte: TOTALCAR (2016)

## 2.4 RASPBERRY PI

O *Raspberry Pi* é um computador em miniatura baseado no processador ARM. Ele pode realizar a maioria das tarefas que um *desktop* realiza, como por exemplo executar planilhas de cálculo, editores de texto e jogos (NEW IT LIMITED, 2016). Segundo Raspberry Pi Foundation (2016), ele foi desenvolvido para permitir que pessoas de todas as idades possam explorar a computação, aprender a programar e entender o funcionamento dos computadores. Na figura 8 observa-se o aspecto da placa *Raspberry Pi 3 Model B*.

Figura 8 – Características do *Raspberry Pi 3 Model B*



Fonte: FILIPE FLOP (2016)

New IT Limited (2016) apresenta a seguinte especificação técnica da placa *Raspberry Pi 3 Model B*:

- computador de placa única com *chipset* Broadcom BCM2837;
- processador *quad core* ARM Cortex-A53 de 1,2GHz;
- 1GB de memória RAM, *Random Access Memory*;
- 40 pinos de GPIO, *General Purpose Input/Output*;
- conexão *Bluetooth* 4.1 integrada;
- conexão *WiFi* 802.11n integrada;
- 1 porta *Ethernet* 10/100;
- 4 portas USB;
- 1 conector combinado de 4 pólos para saída de áudio estéreo e vídeo composto;
- 1 saída HDMI, *High Definition Multimedia Interface*;
- 1 porta CSI, *Camera Serial Interface*;
- 1 porta DSI, *Display Serial Interface*;

- m) 1 porta micro SD, *SanDisk*, para carga do sistema operacional e armazenamento de dados;
- n) 1 porta micro USB para fonte de alimentação.

## 2.5 TRABALHOS CORRELATOS

A seguir serão apresentados dois trabalhos correlatos ao proposto. O item 2.5.1 apresenta o pyOBD, uma ferramenta de diagnóstico automotivo compatível com OBD2 desenvolvida em linguagem de programação Python (PYOBD, 2016). O item 2.5.2 apresenta o EnviroCar, um aplicativo que permite compartilhar informações obtidas através da porta OBD2 (ENVIROCAR, 2016).

### 2.5.1 PYOBD

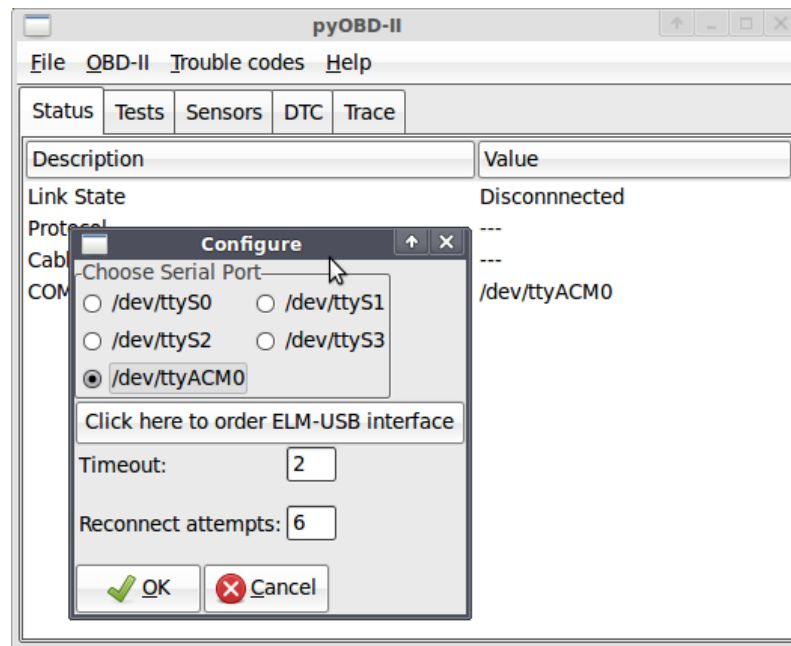
Trata-se de uma ferramenta open source de diagnóstico automotivo, segundo PYOBD (2016), a ferramenta foi projetada para se conectar à porta OBD2 através de uma interface ELM327 USB. PyOBD é voltado para desenvolvedores Python, é composto de um único módulo, chamado `obd_io`, que permite um controle de alto nível sobre os dados dos sensores e gerenciamento dos códigos de erro (PYOBD, 2016). De acordo com PYOBD (2016), o módulo foi testado para funcionar em sistemas operacionais Microsoft Windows, Linux e Mac OSX. Seus pré-requisitos são:

- a) uma interface ELM327 USB;
- b) python 2.x ou superior;
- c) pacote `py_serial`;
- d) um veículo que implemente o padrão OBD2.

Com o pyOBD é possível:

- a) conectar-se ao veículo, conforme a Figura 9;

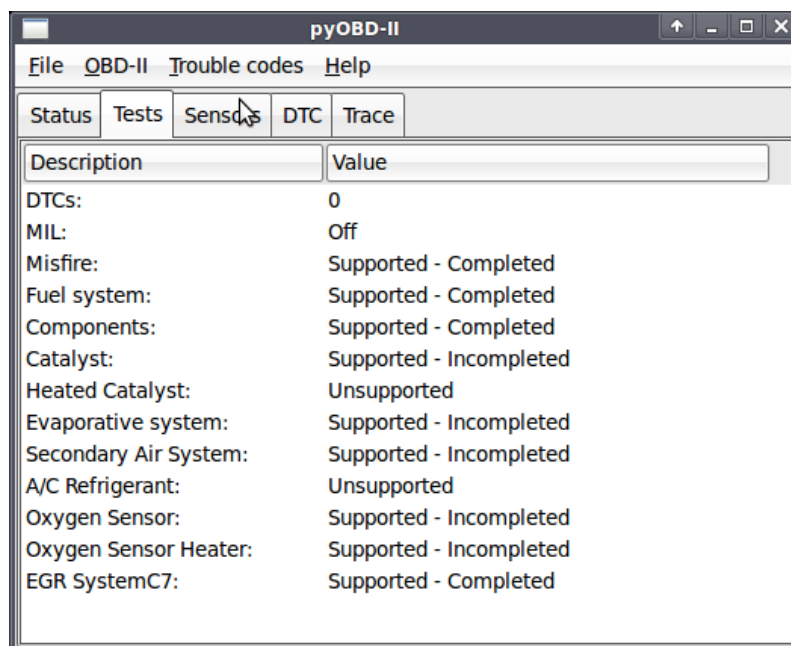
Figura 9 – Conectando pyOBD com o veículo



Fonte: PYOBD (2016)

b) exibir resultados de testes, conforme Figura 10;

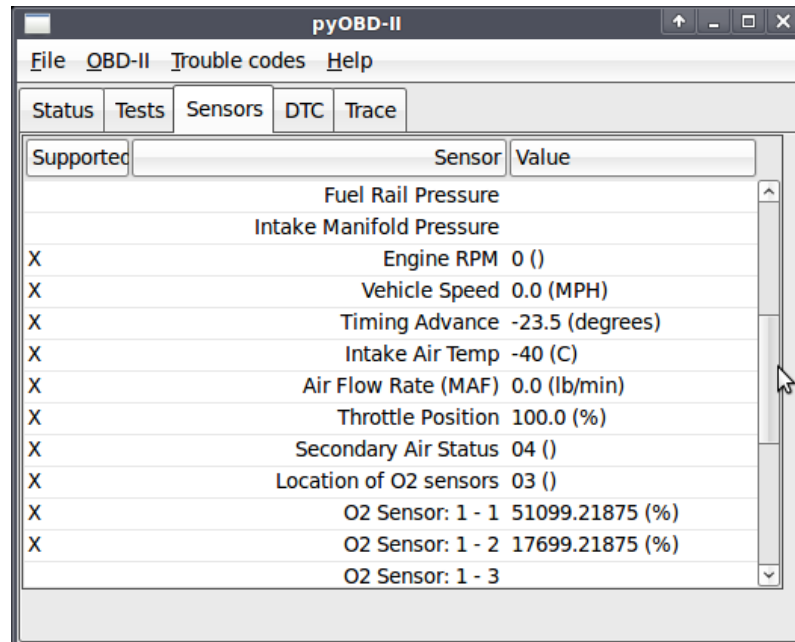
Figura 10 – Exibindo resultados de testes com pyOBD



Fonte: PYOBD (2016)

- c) verificar dados dos sensores em tempo real, conforme Figura 11;

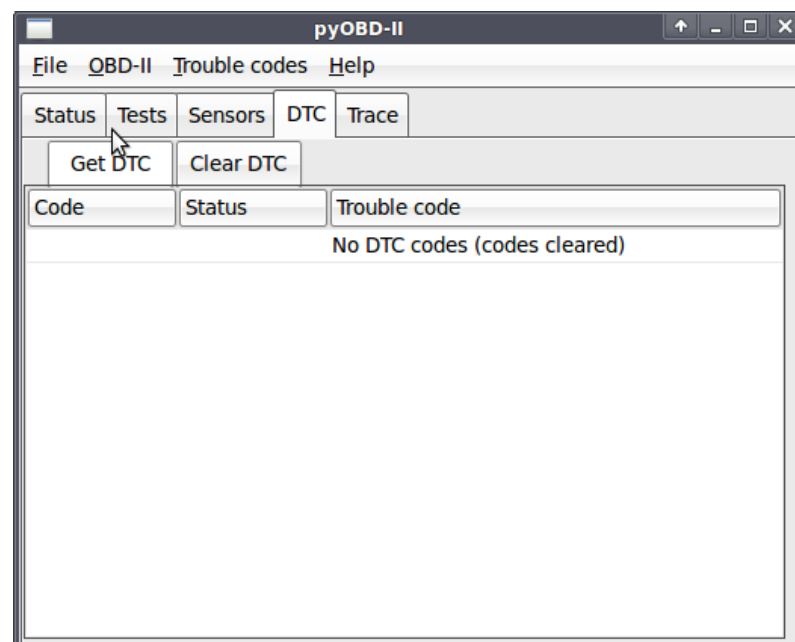
Figura 11 – Verificando dados em tempo real com pyOBD



Fonte: PYOBD (2016)

- d) ler e limpar códigos de falhas DTC, conforme Figura 12.

Figura 12 – Lendo e limpando códigos de falhas



Fonte: PYOBD (2016)

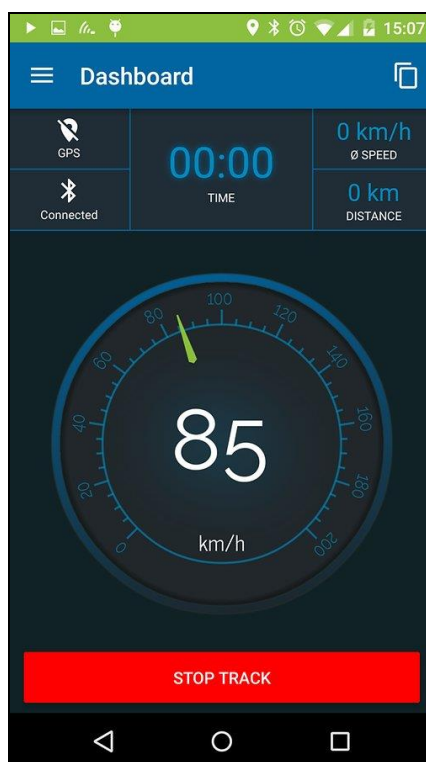


### 2.5.2 ENVIROCAR

Trata-se de um aplicativo alemão open source, desenvolvido para *smartphones* Android, seu propósito é que cidadãos, cientistas, engenheiros de tráfego e indústrias analisem dados OBD2 e compartilhem suas descobertas (ENVIROCAR, 2016, tradução nossa). O Aplicativo se conecta à porta OBD2 através de uma interface ELM327 *Bluetooth*. O usuário pode fazer *upload* das informações obtidas pelo aplicativo, diretamente para o servidor do enviroCar. Segundo ENVIROCAR (2016), os dados ficam disponíveis anonimamente para que cientistas ou especialistas em tráfego acessem estes dados e os utilizem para solucionar questões ambientais e de mobilidade. EnviroCar permite que o usuário perceba o impacto ambiental causado pela forma de dirigir, investigando os dados dos sensores como consumo de combustível, emissão de gás carbônico e de ruídos (ANDROID PIT INTERNATIONAL, 2016).

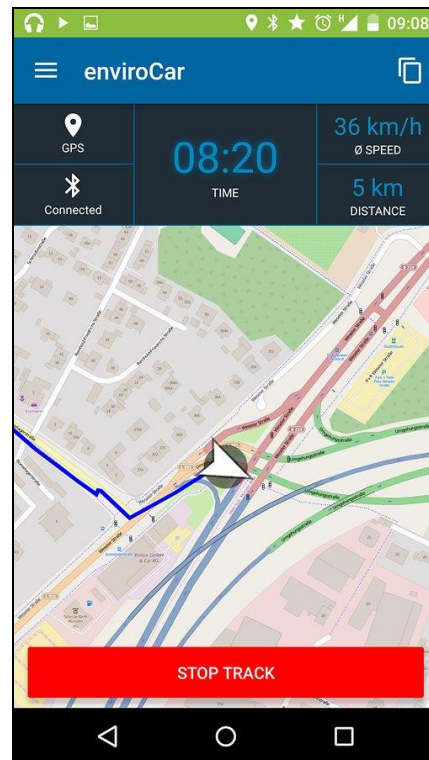
A seguir algumas telas do aplicativo enviroCar. Na Figura 13 observa-se a velocidade do veículo em quilômetros por hora. A Figura 14 é apresenta um mapa com o desenho do trajeto percorrido, o tempo da viagem, a distância percorrida e a velocidade média. Na Figura 15 são apresentadas diversas informações coletadas durante o percurso: marca e modelo do veículo, data e hora de início e término da viagem, consumo de combustível e a emissão de gás carbônico.

Figura 13 – Velocidade do veículo



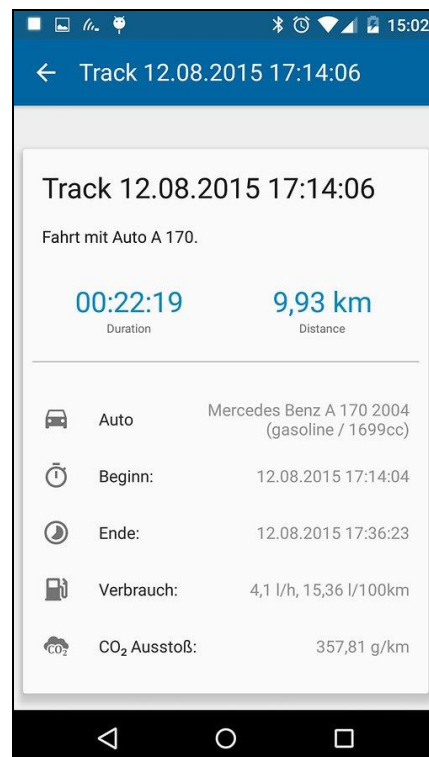
Fonte: ANDROID PIT INTERNATIONAL (2016)

Figura 14 – Velocidade média, trajeto e distância percorridos



Fonte: ANDROID PIT INTERNATIONAL (2016)

Figura 15 – Diversas informações coletadas durante o percurso



Fonte: ANDROID PIT INTERNATIONAL (2016)

### 3 REQUISITOS DO PROTÓTIPO A SER DESENVOLVIDO

Para simplificar a legibilidade, na descrição dos requisitos será utilizado o termo “*firmware*” para referenciar o *software* executando na placa *Raspberry Pi* instalada no carro e o termo “servidor” para referenciar o *software* executando no servidor de aplicações TomCat. Os requisitos do protótipo a ser desenvolvido são:

- a) o *firmware* deverá ser inicializado automaticamente ao ligar o equipamento (requisito funcional RF);
- b) o *firmware* deverá se conectar à porta OBD2 através de uma interface ELM327 *Bluetooth* (RF);
- c) o *firmware* deverá ler os dados da porta OBD2 e armazená-los localmente (RF);
- d) o *firmware* deve tentar estabelecer uma conexão com o servidor a cada 5 minutos (RF);
- e) o *firmware* deverá, ao conectar com o servidor, enviar o número do chassi do carro e os dados OBD2 armazenados localmente desde a última conexão bem sucedida (RF);
- f) o *firmware* deverá ser desenvolvido utilizando tecnologia Java SE (requisito não funcional RNF);
- g) o *firmware* deverá executar em uma placa *Raspberry Pi* (RNF);
- h) o servidor deverá responder às requisições HTTP, *Hypertext Markup Language*, através dos métodos *get* e *post* (RF);
- i) o servidor deverá persistir os dados recebidos do *firmware* (RF);
- j) o servidor deverá persistir os dados em arquivos XML, *Extended Markup Language*, sem a necessidade de utilizar banco de dados (RNF);
- k) o servidor deverá dispor uma página *web* para consultar os dados OBD2 a partir do número do chassi do carro (RF).
- l) o servidor deverá ser desenvolvido utilizando tecnologia Java EE (RNF);
- m) o servidor deverá executar no servidor de aplicações Apache TomCat (RNF);
- n) a página *web* deve apresentar os dados em forma de gráficos (RF);
- o) a página *web* deve apresentar um relatório dos dados em forma de tabela (RF);
- p) a página *web* deve ter interface responsiva de modo que possa ser visualizada em smartphones (RNF);
- q) a página *web* deverá ser desenvolvida utilizando HTML, CSS, *Cascading Style Sheets* e JavaScript (RNF).

#### 4 CONSIDERAÇÕES FINAIS

Conforme *Resolução...* (2004), a padronização do sistema OBD2 nos veículos representa um expressivo avanço tecnológico, possibilitando ao usuário do veículo, através da análise dos dados, prevenir a ocorrência de avarias aos sistemas do veículo, contribuindo para a melhoria da qualidade ambiental, resguardando os interesses da sociedade. Abordando assuntos e tecnologias como Internet das Coisas, OBD2, *Raspberry Pi* e *Bluetooth*, o trabalho proposto tende a enriquecer o material disponível à estudantes de Ciência da Computação, Sistemas de Informação. O trabalho proposto abre a possibilidade de trabalhos de extensão como por exemplo um sistema especialista capaz de analisar os dados OBD2 obtidos pelo protótipo de *software* embarcado e diagnosticar antecipadamente problemas nas no veículo que possam futuramente se transformar em avarias.

## REFERÊNCIAS BIBLIOGRÁFICAS

ANDROID PIT INTERNATIONAL. **Android Apps – Lifestyle**. 2016. Disponível em: <<https://www.androidpit.com/app/org.envirocar.app>>. Acesso em: 18 maio 2016.

ELM. **OBD to RS232 Interpreter**. 2016. Disponível em: <<http://www.elmelectronics.com/DSheets/ELM327DS.pdf>>. Acesso em: 18 maio 2016.

ENVIROCAR. **Off we go**. 2015. Disponível em: <<http://envirocar.org>>. Acesso em: 18 maio 2016.

FILIPE FLOP. **Componentes Eletrônicos**. 2016. Disponível em: <<http://blog.filipeflop.com/embarcados/saiu-o-raspberry-pi-3.html>>. Acesso em: 18 maio 2016.

MANAVELLA, Humberto J. **Diagnóstico Automotivo Avançado**. 3. ed. São Paulo: HM Autotrônica, 2009. p. 121-127. Disponível em: <<http://www.hmautotron.eng.br/zip/cap19-hm004web.pdf>>. Acesso em: 18 maio 2016.

NEW IT LIMITED. **RPi 3 (2016) Model B**. 2016. Disponível em: <[https://www.newit.co.uk/shop/all-raspberry-pi/raspberry\\_pi\\_3/raspberry\\_pi3](https://www.newit.co.uk/shop/all-raspberry-pi/raspberry_pi_3/raspberry_pi3)>. Acesso em: 18 maio 2016.

NG, Audrey. **THE EVOLUTION OF THE “INTERNET OF THINGS”: FROM “DIAGNOSTICS AND REPAIR” TO “PRESCRIPTIVE AND PROACTIVE”**. 2015. Disponível em: <<http://br.hortonworks.com/blog/the-evolution-of-the-internet-of-things-from-diagnostics-and-repair-to-prescriptive-and-proactive>>. Acesso em: 18 maio 2016.

OUTILS OBD FACILE. **Automotive Electronic Diagnostic**. 2016. Disponível em: <<http://www.outilsobdfacile.com/obd-mode-pid.php>>. Acesso em: 18 maio 2016.

PYOBD. **Open Source OBD2 diagnostics**. 2015. Disponível em: <<http://www.obdtester.com/pyobd>>. Acesso em: 18 maio 2016.

RASPBERRY PI FOUNDATION. 2016. Disponível em: <<https://www.raspberrypi.org>>. Acesso em: 18 maio 2016.

**RESOLUÇÃO CONAMA nº 354, de 13 de dezembro de 2004**. Publicada no D.O.U. nº 239, de 14 de dezembro de 2004, Seção 1, p. 62-63. Disponível em: <[http://www.mma.gov.br/port/conama/legislacao/CONAMA\\_RES\\_CONS\\_2004\\_354.pdf](http://www.mma.gov.br/port/conama/legislacao/CONAMA_RES_CONS_2004_354.pdf)>. Acesso em: 18 maio 2016.

RIORAND. **Advanced Technology**. 2015. Disponível em: <<http://www.riorand.com/on-board-diagnostics>>. Acesso em: 18 maio 2016.

SAE INTERNATIONAL. **The ultimate knowledge source for mobility engineering**. 2016. Disponível em: <[www.sae.org](http://www.sae.org)>. Acesso em: 18 maio 2016.

**THE BEST OBD2 SCANNERS**. 2016. Disponível em: <<http://thebestobdiiscanners.com/10-modes-of-operation-for-obd2-scanners>>. Acesso em: 18 maio 2016.

TOTALCAR. **Diagnostics Support**. 2014. Disponível em: <<http://www.totalcardiagnostics.com/support/Knowledgebase/Article/View/72/15/elm327-review--about-elm-327-obd2-interface>>. Acesso em: 18 maio 2016.

ZURAWSKI, Richard. **Automotive Embedded Systems Handbook**. Florida: CRC Press, 2009. p. 33-34.