Original Research Paper

A Low-Cost Vehicular Infrastructure of Monitoring and Communicating Information for Intelligent Transport Systems

Daniel Rodrigues, Jessica Nayara Severino, Andre Luis Cristiani, Fabiano Tadeu Mathias Costa, Luis Hideo Vasconcelos Nakamura and Rodolfo Ipolito Meneguette

Federal Institute of São Paulo, Catanduva, Brazil

Abstract: Currently, vehicles have a very large range of sensors and actuators, capable of not only sensing the condition of vehicles but also the environment around them. The information from these sensors can be used for various services in which they can bring greater comfort, safety and entertainment to the passengers and conductors of the vehicle. However, the great challenge lies in the abstraction and propagation of this information to a service center or to an application in a secure and accessible way to all users. Thus, with the objective of developing low-cost information abstraction devices accessible to all, we propose the development of an architecture that will perform the capture of vehicular data through an electric circuit proposed, as well as the propagation of this information through a web service of monitoring school vans that can be accessed through an application focused on smartphones. To verify the functioning of this architecture we use our monitoring of school vans system as a case study.

Keywords: Embedded Device, Vehicle Device, Vehicular Network

Introduction

The automotive industry has been heavily investing in new vehicle-mounted devices with the aim of improving vehicle safety, comfort, stability and performance Xie *et al*. (2017); Meneguette *et al*. (2016). Thus making the electronic system of a vehicle more complex, considering not only the functionality of these components but also its architecture. Studies Tuohy *et al*. (2015) describe that the number of Electronic Control Units (ECUs) in a standard car has gone from 20 to more than 100 in the last decade. In addition, there was an increase in the number of software embedded in the vehicle, from 2% to 13% of the total value of a vehicle and the number of lines of code increased from one million to over 10 million between the years 2000 and 2010 Charette (2009). An example is an increase in the functionality and devices on board the vehicle and the emergence of self-propelled vehicles using a variety of technologies to allow the movement of vehicles in the direction of a human being. An example of a self-propelled vehicle is the S500 Intelligent Drive, produced by Mercedes Benz, which was able to travel a distance of 100 km without human intervention. This self-directed demonstration involved dealing with rotations, traffic jams, traffic signals and pedestrians. The vehicle could react to these diverse obstacles by means of a camera, radar and geolocation system Euronews (2017).

Electronics in current vehicles are responsible for executing pre-programmed and specific functions, based on input data Meneguette and Boukerche (2017); Amarasinghe *et al*. (2015). These data are generated from several sensors and then transmitted to the electronic control units, enabling the management and guaranteeing the good operation of the vehicle, as well as the safety of the users Li *et al*. (2016); Zhang *et al*. (2017). These various electronic units communicate through standardized messages according to each assembler, taking as reference a communication protocol created by the company Bosch and denominated Controller Area Network (CAN) Farsi *et al*. (1999). The information obtained by the sensors helps the driver to identify any defects in the vehicle when they are shown on thevehicle's technical support, proposing a greater precision and agility. To access the fault codes, which originate when a system malfunction is detected, a diagnosis connector must be inserted in the vehicle. In

this context, the Society of Automotive Engineers (SAE) Society of Automotive Engineers (1997) has regulated the self-diagnostic On-Board Diagnostics (OBD), a standard adopted in all automotive vehicles sold since 1996 in the United States and Europe. In Brazil, this mandatory system was regulated in 2010 Machado and Oliveira (2007).

With the requirement for an OBD-II system in vehicles, the interest in new technologies that could interact with the new system has grown. Connectors with serial or Bluetooth interfaces have started to appear on the market allowing more comprehensive access to the information collected by the self-diagnostic system, such as information on the operating status of various sensors, actuators and electronic units available in the vehicle Marin *et al*. (2017); Tuohy *et al*. (2015). As well as the fault codes in the line of communication and faults of the electronic system registered in the electronic units. With the abstraction of these data and the concern for the safety of people using vehicles every day it has become attractive to create an application that can inform the user the actual state of your vehicle and the safety, it can provide you with regard to its basic features Farrugia *et al*. (2016).

In the literature, there are works proposing an architecture that uses several diagnostic devices for abstraction of the information with the objective of offering new services to the vehicles. However, some of these architectures are not available Arada (2018), neither the diagnostic system circuit nor the form of connection to the central service controller (web service or cloud). Other works Pereira *et al*. (2016); Farrugia *et al*. (2016); Xie *et al*. (2017); Jan *et al*. (2014); Amarasinghe *et al*. (2015); Wei and Hong-ying (2016); Wang *et al*. (2018) use a mobile device to perform the communication between the central service controller and the diagnostic device, thereby increasing the delay in communication between the data captured by the monitoring device with the users who are consuming services that depend on this information.

In view of the limitations found in current devices and in order to offer low cost and easy access technology to all, the present work proposes the development of a new diagnostic device capable of transmitting information directly to the service center giving greater agility in spreading the information on the vehicle. In addition, the device will also provide communication to mobile devices so that the user can manage the state of their vehicle. We also propose the development of a Web service as well as a school vans monitoring service as a case study to verify the operation of the developed device. Therefore, the major contribution of this work is the development of a low-cost device which is accessible to all to carry out monitoring of vehicle sensors as well as communication with a central service, capable of analyzing and processing the information for possible decision-making.

The present work is structured in five sections. Section 2 presents the works of the literature. Section 3 proposes an architecture focused on the abstraction of vehicle information and the dissemination of such information to the vans monitoring service and conducts a case study considering the school vans monitoring service in the city of Catanduva-SP (Brazil) which is exposed in section 4. The section 5 presents the conclusion and the future work.

Related Work

Both in the literature and in the market Solutions (2018); Electronics (2018); Arada (2018) there are some devices that read the sensors embedded in the monitoring vehicle of the information that travels in the internal network, the CAN network. Many of these devices use Bluetooth communication, as is the case with the device developed by ELM327 Electronics (2018), such device communicates with a mobile phone or tablet through an application that encodes of the data received by the sensors. Other devices available on the market are those developed by the company ARADA Arada (2018). These devices, in addition to scanning the information of sensors embedded in the vehicle, also offer all communication among vehicles via the 802.11p protocol and a road infrastructure using the LTE communication and also 802.11p - Wifi. However, these devices have a relatively high price and are proprietary, i.e., manufacturers do not provide the form in which these devices are implemented. In addition, the infrastructure for storing and propagating information captured by the sensors is not available either.

In the literature, some studies use these diagnostic devices to provide new services for intelligent transport systems. For example, the work of Pereira *et al*. (2016) used an OBD to capture vehicle information with the purpose of inferring the fuel consumption based on the behavior of the driver. In order to identify the behavior of the driver, vehicle data are obtained through the OBD-II standard. The data provided by the mobile device is captured by the smartphone application and converted to logical fuzzy, whose rules and sets have been developed in conjunction with a specialist. With knowledge of how to consume, the driver can adapt his driving habits in order to reduce the fuel consumption of his vehicle.

Hilpert *et al*. (2011) OBD-II is used as well as data from GPS to calculate the production of carbon dioxide during the journey of the vehicles in the roads. To do this, the system has a mobile device that receives OBD information and passes this information to a central office, which performs the production calculation carbonic. Therefore, the proposed system calculates CO_2 emissions from the vehicle based on the consumption of gasoline captured by OBD and the distance traveled by the vehicle captured by GPS. Such a solution as the above uses the mobile device as

an interface between the sensor reader boarded on the vehicle and an information processing center.

In addition to the use of passenger cars, this strategy can also be used in commercial freight and public transport vehicles. In the work of Farrugia *et al.* (2016) describes the use of OBD for the abstraction of diesel oil consumption in heavy vehicles. With the information on fuel consumption, the authors verified the quantities of Nox and O2 emitted into the atmosphere. This solution also used a mobile device to communicate with the OBD and processed the available data to inform the user of the percentages of each gas that the vehicle was emitting.

Although these works use a proprietary device for reading the sensors, which does not allow the development of new circuits to aid in the communication of information with a service center, other work focusing on the development and expansion of such information retrieval devices. As, for example, another work have developed a device for carrying out the collection of information from the sensors boarded in the car Xie *et al.* (2017). This device has enabled the communication and the propagation of the information captured. The use of Wifi technology does not allow the propagation of information over long distances, making it difficult to transmit information obtained to a service center.

Another work proposed a device that would read the sensors embedded in the vehicle as the communication with the location service (GPS) Pal and Pal (2017). For this, the authors used ready-made platforms to perform all the processing of the captured information which is transmitted over a wireless network, as well as over a wired network. Due to the transmission power, this device has a low transmission, making it difficult to connect directly to a service center.

This work proposes the development of a device for the OBD, which provides direct communication with a service center (cloud or datacenters). low-cost system capable of connecting to a storage and information propagation service and also allowing the user to access the state of his vehicle in real time by means of a mobile device. This work also proposes the development of storage and available infrastructure for new services that use the information abstracted by sensors embedded in a vehicle.

A Low-cost Vehicular Infrastructure

This section presents the development of a complete infrastructure for abstraction, storage and processing of sensor information that is embedded in the vehicle. This work proposes an architecture called ICCV – Infrastructure for Capturing and Communicating the Vehicle, which aims to provide services and information to optimize transport and access relevant information not only of the vehicle, but also of the traffic of vehicles on the roads.

This architecture is composed of 3 basic components: (i) Services - responds to the processing of the abstracted information to the aid of the transport system, providing applications for optimization and monitoring of traffic. (ii) Storage and propagation of information - responds by storing, analyzing and propagating the information of the sensors on the vehicle in order to provide information that will be used by several services. (iii) Reading devices corresponds to the development of new devices capable of reading the information on board the vehicle and also allowing the abstraction of new information. Thus, it provides a complete infrastructure to abstract, store and make available information and services not only for drivers but also for all citizens. The Fig. 1 shows the 3 main components.

Service

The service component will demonstrate interaction with the end-user, as well as control of the entire infrastructure for capturing and processing information. This component is subdivided into 3 modules as shown in Fig. 2 for the execution and development of these modules, the web concept and the HTML 5 functionality were used to allow any device to have access to the services:

- Visualization: Responsible for showing the main information to the user and allowing the user to interact with the system. For example, the visualization of the location of the school van carrying a particular student
- Request and response processing: responsible for managing the requests and responses of the user's interactions with the set of services provided. For example, the request regarding the distance and time of arrival of a school van until school. In this case the system will respond with the location of the van and the time of arrival of this van
- Communication: responsible for carrying out the communication with the component for storing and propagating information in order to allow the abstraction of the information requested by the service. This module establishes a secure connection through the use of SSL. This component is directly linked to the storage and propagation of information, using the concept of web service

Storage and Propagation of Information

The storage and propagation component will provide storage tools for information from sensors. In addition, it will provide a contextualization and processing of such information, such as gasoline consumption, route traveled, number of vehicles, location, among others, as can be seen in Fig. 3. This component also disseminates this information to the various services that the users are

executing. Therefore, this component can be divided into 3 main functionalities:

- Storage: which will store the information sent by the reader device of the sensors embedded in the vehicle. For this, a relational database, Mysql was used for the storage of GPS information, route, fuel consumption among others
- Processing: it brings together various information to carry out the contextualization, as well as the analysis of the conditions of the routes. For example, in order to estimate the arrival of the van to school, it is necessary to use the geographic position of the van and the school, the average speed of the van and the condition of the traffic of the streets that are part of the route. Thus, with this information it is possible to estimate the time a van will reach its destination communication: responsible for communicating with the reading device in order to obtain the information

For the storage and later for the information processing, the Firebase (2018) was used, in which it offers tools and infrastructure for developers that help in the development of connectivity between heterogeneous devices. Firebase is a cloud computing service that provides an efficient platform for connecting the information capture device to mobile and Web applications from APIs and SDKs, saving development time. This service allows the server to send messages to its clients through notifications. This tool was used because it will provide greater scalability for the system as well as allow greater heterogeneity in the devices that will consume the proposed services.

The functionalities provided by Firebase and used in this work were Authentication in which they allow the integration with several social networks to perform the authentication of the users in diverse devices. So, by registering a new Firebase user, the tool will create a Firebase User that contains a unique ID that will reference the selected platform associated with other user data. The integration adopted in this work to carry out the authentication was the combination of e-mail and password.
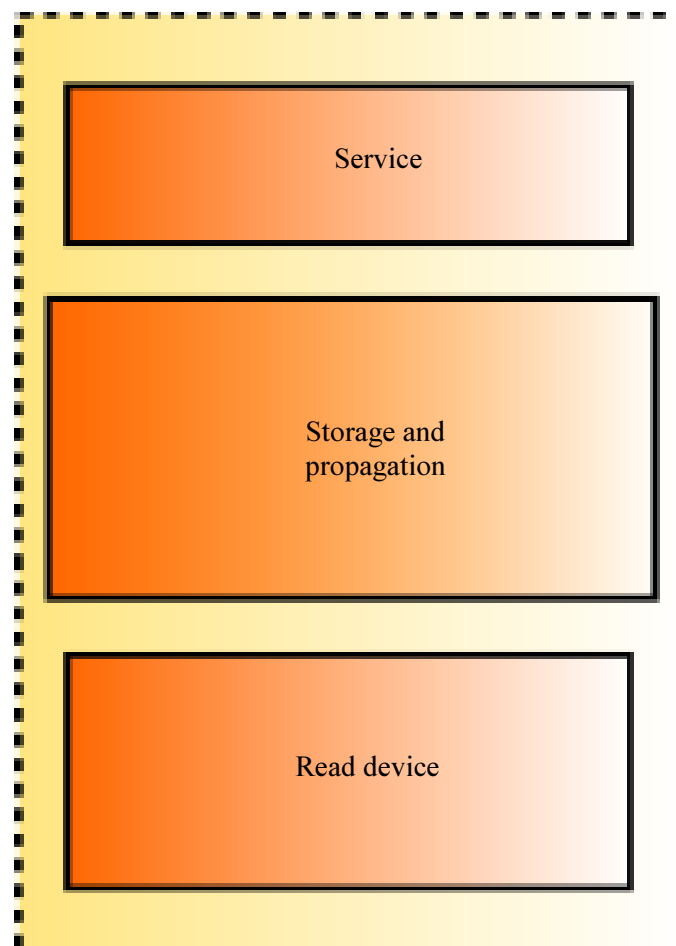

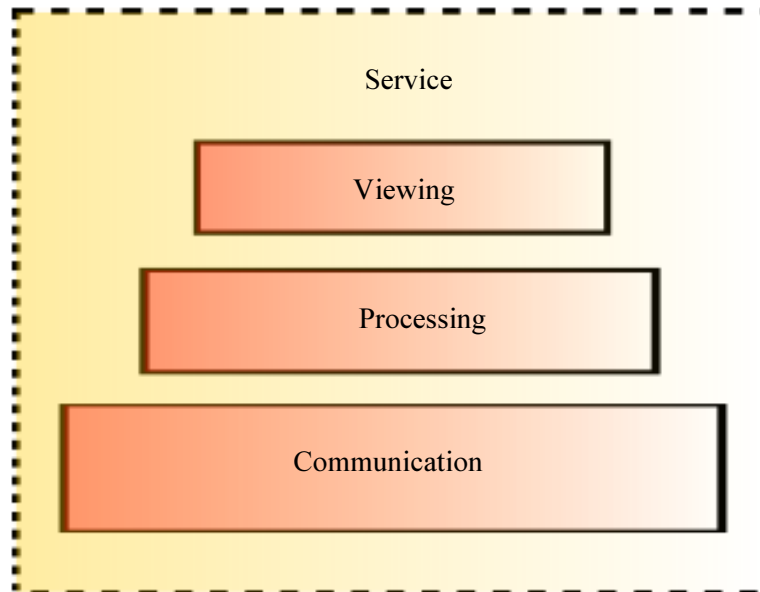
Fig. 1: Architecture ICCV and its main components

933
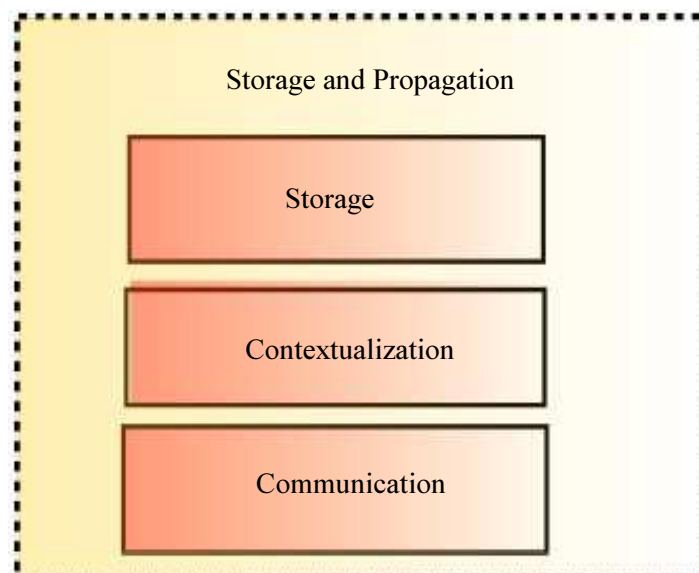
Fig. 2: Service component abstraction

Fig. 3: Abstraction of the Information Storage and Processing component

Firebase also uses the Realtime Database tool in which it allows you to structure the data to be stored in the database in JavaScript Object Notation (JSON) formats and as data is updated or entered, all client applications are updated instantly. The main factors that influenced the choice of this database were: its scalability potential, speed of update and the synchronization of the data in the clients devices in real time, which minimized work in the development of data synchronization in the application. The JSON structure of the application database was as follows:

```
1 "driver" :
2 {
3 "driver's key : { "email" : "", "
      keyLocation" :
4 "key of the driver location node", "name"
      : ""
5 }
6 },
7 "key of the driver location node" : {
8 "latitude" : , "longitude" :
9 },
10 "school" : {
```

```
11 "school's key" : { "address" : "", "email"
      : "", "name" : "" }
12 },
13 "user" : { "user's key" : {
14 "address" : "", "email" : "", "name" : "",
      "son" : {
15 "driver" : {
16 "email" : "", "keyLocation" : "", "name" :
      "" }, "name" : "",
17 "school" : {
18 "address" : "", "name" : "" }
```

Algorithm 1: Database abstraction

For the capture and localization monitoring were used latitude and longitude information of the van in which they were later placed in the Google Maps for the visualization of the van, to carry out the calculation of the estimated arrival of the van in the school and in the house of the student. For this Google Maps uses an API called Google Maps Directions in which allows you to manipulate geolocation information. The requests to the Google Maps Directions API are performed through an HTTP interface using the GET verb. The request URL must be assembled by carrying text strings or latitude and longitude coordinates to identify the locations along with the API key.

Reading Device

For the abstraction of vehicle information a new device called ORD - OBD Readout Device is responsible for allowing the storage and propagation component to receive the information captured directly without any intermediate device. For this, ORD has a mechanism for communicating with the cellular network, as well as short-range communication via Bluetooth to allow the user to also monitor the status of his vehicle.

Another component used in this project was the MCP2515, it is an 8-bit Controller Area Network (CAN) from Microchip Technology that has specification to operate in version 2.0 B of the protocol. Its operating voltage is from 2.7V to 5.5V at speeds up to 1 Mb/s. The MCP2515 performs and modulates the network signal according to the control of protocol mechanisms, masks, filters, transmissions and buffers. It is also responsible for the records that are used to configure a device in its operation.

Thus, the ORD has a MCP2551 component which will carry out the conversion of the differential signal of the CAN network to a digital level logic signal and the reverse. The MCP2551 is a differential transmitter and receiver of the CAN protocol that has fault tolerance. This device links and interfaces between the CAN protocol controller and the data bus. It has an operating voltage between 4.5V to 5.5V, speed up to 1 Mbs and is within the ISO-11898 standard.

In addition to the transceiver and the network controller, a microcontroller from the company Microchip model ATmega 328P was also used. It is a family of 8-bit CMOS (complementary metal-oxide-semiconductor) microcontrollers based on the AVR architecture initially launched by ATMEL, which has 32 KB of memo To store code, besides 2 KB of SRAM and 1 KB of EEPROM (Atmel, 2016). This microcontroller has a range of 1 MIPS/MHz (1 Million Instructions per Second per Mega Hertz) with 32 K of Flash memory. Its operation voltage is 1.8V to 5.5V and has 23 individually programmable inputs and outputs.

Also used was the Voltage Regulator 7805, which is an Integrated Circuit (IC) capable of regulating the output voltage. It can receive from 7V to 20V and provides a stable voltage of 5V and a maximum current of 1A.

Use Case

As proof of concept of the infrastructure was developed a prototype of a monitoring for school in the city of Catanduva. This system allows the reading of the sensors embedded in the van and performs the processing of this information, uses the average speed information of the vehicle and location to estimate the van's arrival time at the school or at the student's home. For this, the reader device of the sensors embedded in the vehicle was first developed. The initial prototype of the ORD follows the pattern of the commercial devices, i.e., an electric circuit was created in which Bluetooth is used to make the communication with the mobile device as well as a USBport to check the information abstracted from the sensors, as shown in Fig. 4.

Fig. 4: Testbed of ORD

935

This initial prototype was developed in the protoboard and the circuit was printed for the verification of the operation. Afterwards, the GPS circuit and the communication interface with the cell phone were added. From the development of prototype hardware for testing, the entire logic of the CAN network protocol was implemented in an algorithm to handle the flow of network data signals. Thus, it was possible to obtain some data in hexadecimal values, relatively similar to the standard structure of the data packets.

From these collected data it was possible to carry out tests with the proposed device connected to the vehicle. For these tests a vehicle manufactured by Hyundai, model IX35 (year 2013, model 2014) was used. The vehicle operated a pre-determined circuit using the dual tracks of Avenida Theodoro Rosa Filho, located in the city of Catanduva-SP. This avenue has a maximum speed limit of 60 km/h.

During the tests, speeds of 40, 50 and 60 km/h were maintained with the fixed gear to collect correlated data and enable their verification. The desired speed of the vehicle was kept constant to be possible to capture the velocity parameter that is used in the case study. Figure 5 describes the set of collected data which the velocity is highlighted in yellow.

These results refer to the verification of the data collected using the collected speed, which was kept constant at 40 km/h. The hexadecimal value -"24", which is highlighted in yellow, represents the speed of the vehicle at the time of data archiving. The speed can be identified due to the hexadecimal base-to-decimal conversion, returning the exact value of the speed of the vehicle at the moment of data acquisition. The speed varied at 40, 50 and 60 km/h and at the desired speed the same was constant and the difference between the value obtained and the value presented on the vehicle speedometer ranged from 5% at higher speeds and 10% in lower speeds, as can be seen in Table 1.

It supports the development and testing of the reader of sensors embedded in the vehicle developed at the service center. Therefore, every time the driver's location is updated in the web service, the firebase releases a notification to all customers informing them that there are divergent data. Every time this notification is received, a new request is made to the Google Directions API by passing the new location of the driver, thereby updating the route data. The Algorithm 2 describes the call of the function listening for new notifications from the web service for a notification update.

Table 1: Checking the information capture considering the vehicle speed

Speedometer	40 Km/h	50 Km/h	60 Km/h
ORD	36 km/h	47 Km/h	57 Km/h

```
ID   DLC Data                              Period Count Comment
0A0   8   7B 81 5C 18 24 2A 4  00    0     557
0A1   8   80 81 00 00 30 00 00 00    0     458
18F   8   FA 46 00 00 00 6E 00 20    0     13218
260   8   0B 39 42 30 01 96 B7 77    0     5491
2A0   8   48 00 8B 1C 67 28 C6 03    0     4206
2B0   5   0B 00 02 07 C2             0     18365
316   8   05 39 70 17 39 15 24 6B    0     19743
329   8   DD B7 78 10 11 2F 1D 14    0     4321
350   8   11 10 D3 94 94 00 00 D2    0     19670
370   8   00 20 40 00 00 00 00 00    0     4531
382   8   00 00 00 00 00 00 00 4C    0     2321
43F   8   44 45 60 FF 74 1E 16 00    0     3999
440   8   FF 04 23 00 FF 1F 16 7F    0     5531
4B1   8   E5 B0 6D F0 00 00 00 F2    0     9285
4F0   8   00 00 00 00 00 EA 6B 08    0     9696
545   8   B8 00 00 8C 00 00 00 00    0     15253
580   8   00 00 1F 11 12 20 01 12    0     5531
59B   8   00 05 00 00 00 00 00 FC    0     2039
5A0   8   00 00 00 00 00 00 00 00    0     101
5A1   8   50 36 52 37 47 41 28 86    0     209
5A2   4   09 00 00 00                0     187
5E4   3   00 00 00                   0     2239
690   8   00 00 00 00 00 00 00 00    0     2077
```

Fig. 5: Data collected in vehicle tests at a speed of 40 km/h

```
1 DatabaseReference myRefLat;
2 myRefLat = database.getReference(user.
       getSon().getDriver().getKeyLocation())
       ;
3 myRefLat.addValueEventListener(new
       ValueEventListener() {
4      @Override
5      public void onDataChange(DataSnapshot
          dataSnapshot) {
6        LocationData locationData =
             dataSnapshot.getValue(LocationData
             .class);
7        sendRequest(locationData, user.getSon
             ().getSchool().getAddress(),
8          user.getAddress());
9      }
10     @Override
11     public void onCancelled(DatabaseError
          error) {
12       Toast.makeText(getActivity(), "Request
             error!",Toast.LENGTH_SHORT).show
             ();
13     }
14 })
```

Algorithm 2: New notifications from the webservice for a location update

Whenever a route is first consulted or updated, the map settings are determined and the route along with the markers representing the driver's current location, school and delivery location are drawn on the map.

For the use of this information carried out the development of two applications (customers) to the system aiming to conduct a complete system test, monitoring of school vans aiming to chart the best trajectories for these vans. For this, we use the positioning the speed of the van, captured by the device developed, with this information associated with the traffic conditions captured by Google Maps, we are able to estimate the arrival time of the van until the student's home or school.

The first customer was developed in which it would allow the school and city hall to manage the transportation linked to both the school and the entire school van system of a city. The web application displays the route in an easy and intuitive way, showing the current position of the van or bus, being represented by an image of a yellow bus, as we can see in the Fig. 6.

The other client covers the visualization of the student's father as well as the profile of the driver of the van that this client was developed for android platform. The Fig. 7 shows the route in the view of the student's responsible, taking into account the current position of the driver of a public school vehicle, the address of the State School *Antônio Maximiano Rodrigues* and the student's residence.

The interaction among the parents of the students or the school responsible with the system is initially done through a web client that allows them to see the route of the van, its location and estimated time of arrival of the van, considering the speed of the vehicle, its route and its location. The profile of the parent or responsible student (Fig. 8) is started when the student enters his/her registered login and password information in his or her register. Before sending the data for verification in Firebase Authentication, the application verifies that the login field and/or password are blank, in this case a message is displayed for the user to fill in the fields. Once that authentication is successful then the application is started displaying the map with the current position of the driver the route to be traveled, the time the driver will take and the distance he will travel to leave the student in his residence.
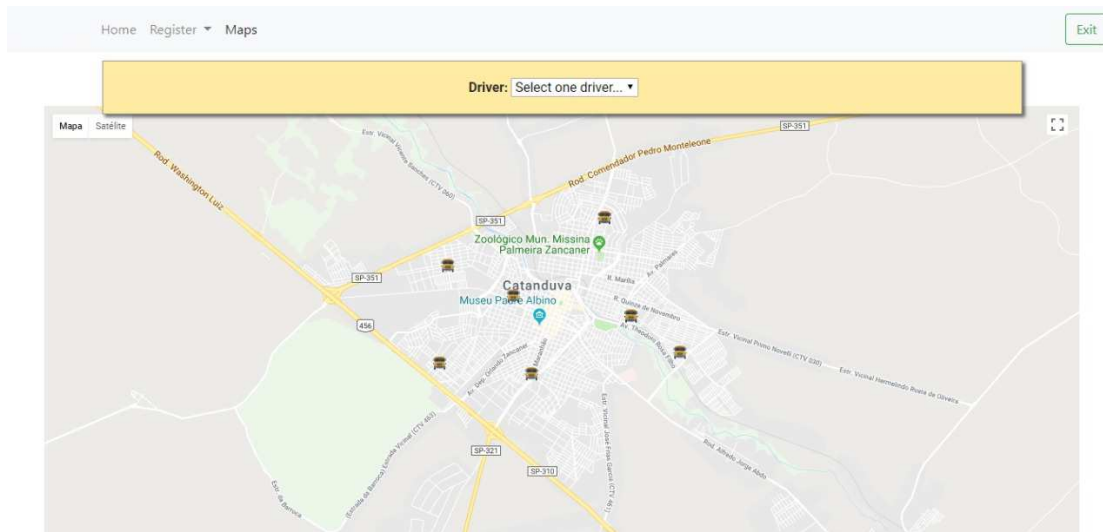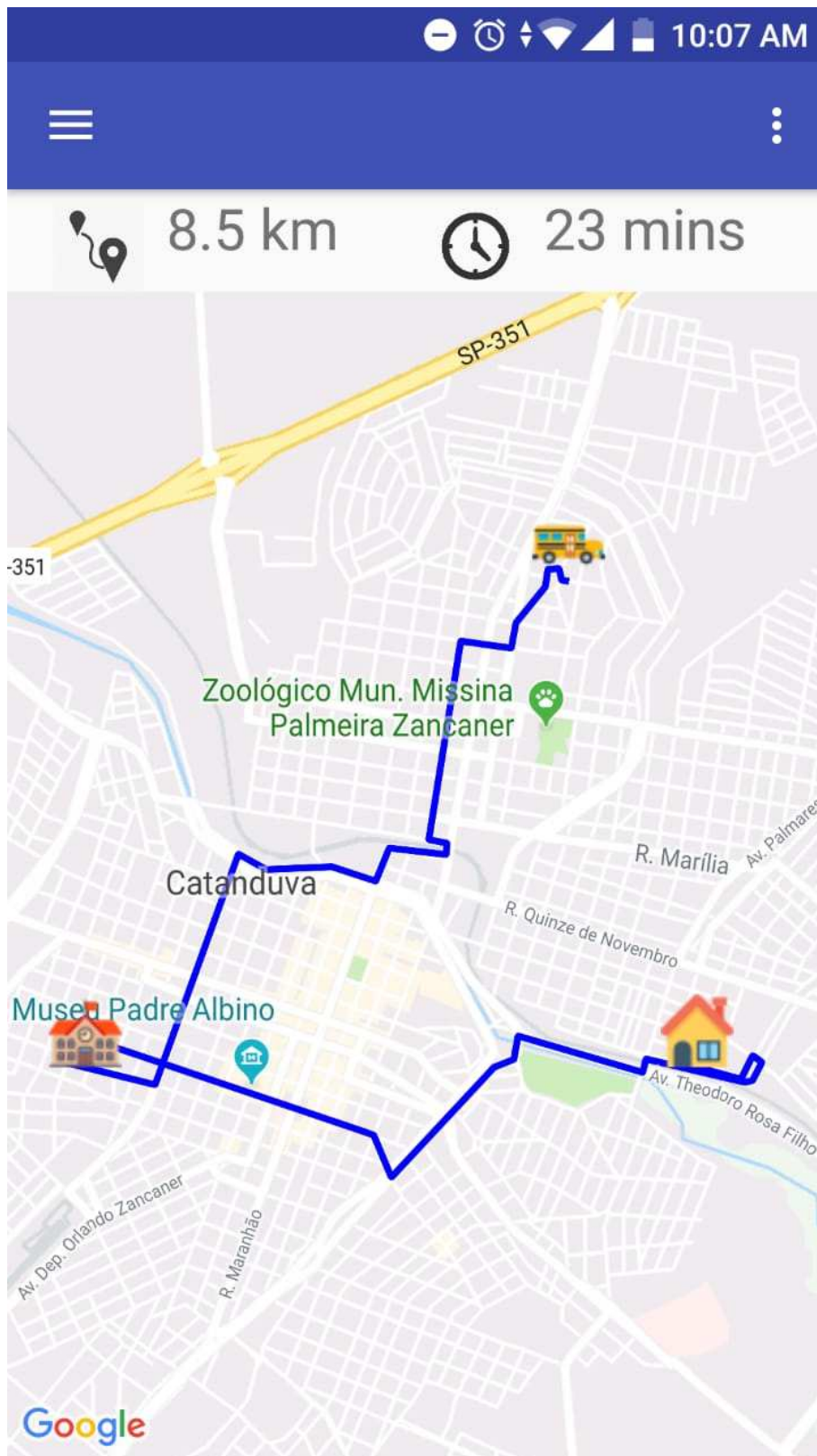

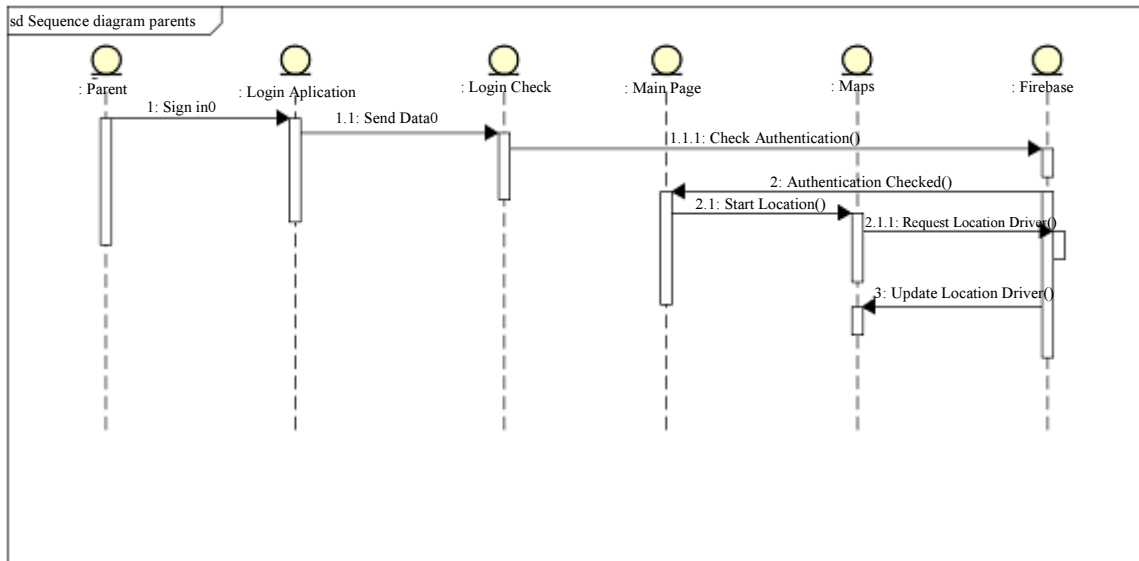**Fig. 6:** Web view

Fig. 7: Android view

938

Fig. 8: Sequence diagram of the parents profile
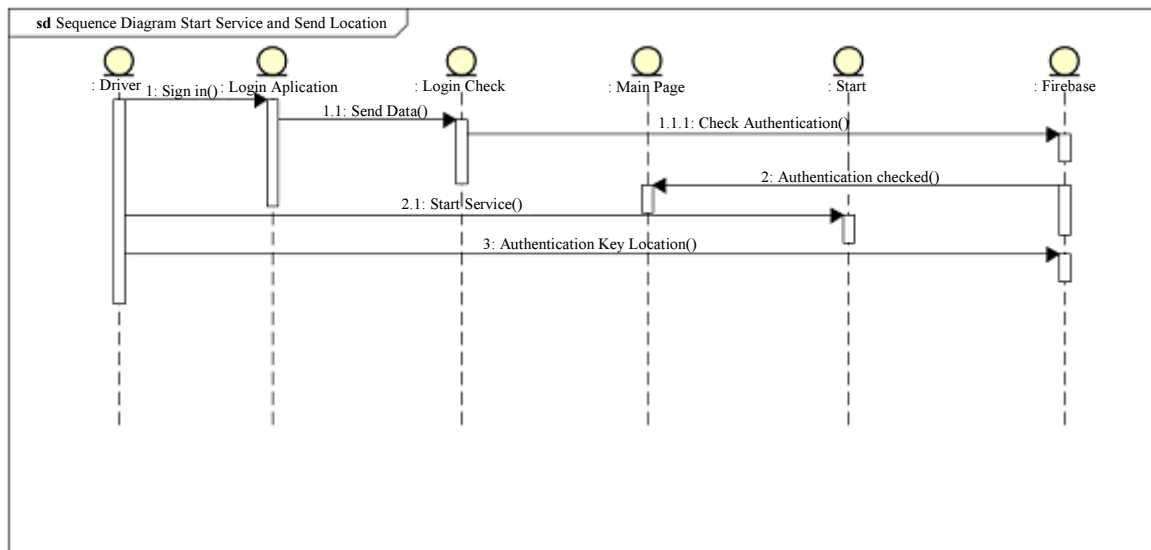

Fig. 9: Sequence diagram of the driver profile

The driver profile (Fig. 9) starts when it enters its login and password information in the application interface, this information goes through a field verification process, in case one of the fields is in blank space, a message is sent to the user to fill in the information correctly. When the information has been validated, it is sent to the Firebase Authentication and verified in the database, if the data is true, the user interface with the Start Service and Stop Service. The Stop Service feature is kept disabled until the driver starts the location service, by clicking on the Start Service it is blocked and the Stop Service is enabled. In order to the driver's location to be sent to the web service, it is necessary for the driver to authorize the application toverify its location, so that this information is sent to the web service and updated in the application of the responsible student, updated on the city's profile and from school. When a driver does not start your application, a message is displayed to the user informing that the user is offline.

Evaluation of User Case

The tests were carried out on two different smartphone models, both with the Android system, the first smartphone (Smartphone 1) has 64 GB of internal

storage android version 7.0, 5.1-inch display, LTE 4G mobile data, Gsm Quad Band network (850/900/1800/1900), quad-core processor 1.5 GHz Cortex-A53 + Quad-core 2.1 GHz Cortex-A57, 64-bit operating system, ARM Mali-T760 GPU, 3 GB RAM, 802.11 a/b/g/ GPS A-GPS/, with maximum download speed of 300 Mbps and upload speed of 50 Mbps, has a Lithium-type battery and capacity of 2600 mAh.

The second smartphone (Smartphone 2) has 64 GB of internal storage android version 8.0, 5.5-inch display, LTE 4G mobile data, network Gsm Quad Band (850/900/1800/1900), Cortex-A53 chipset Qualcomm Snapdragon 625 MSM8953, 2 GHz 8 core processor, 64-bit operating system, Adreno 506 GPU, 4 GB RAM, 802.11 wifi, maximum speed for downloading 150 Mbps and upload speed of 50 Mbps, has a LiPo type battery and capacity of 3080 mAh.

The tests were performed using 4G and Wi-Fi networks. For the use of Wifi, the wireless network mesh infrastructure (802.11 g/n) managed by Catanduva prefecture was used to provide coverage of the main neighborhoods of the city and downtown. Thus, allowing communication between the elements of the ICCV. Tests were observed with both the parent and the driver profiles. When we analyzed the driver's profile, it was analyzed in time to update the positioning, battery and amount of data used. When we evaluated the father's profile we considered the battery consumption and the amount of data used.

Driver Profile

The Fig. 10(a) shows the average time to update the database. We can observe that the response times of the updates are smaller when the 4G data is used because of its greater bandwidth. Due to a low competition for the communication channel due to a low amount of cell phones using such technology. When we compare the two smartphones using the 4G, we can observe that smartphone 1 has a longer time, this is due to the waiting time for the transmission due to internal competition of communication interface with other applications. The same thing happens when using Wi-Fi.
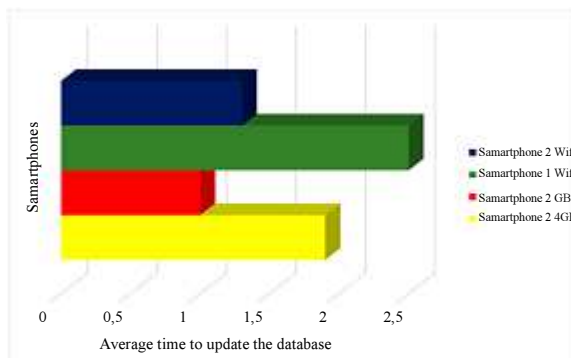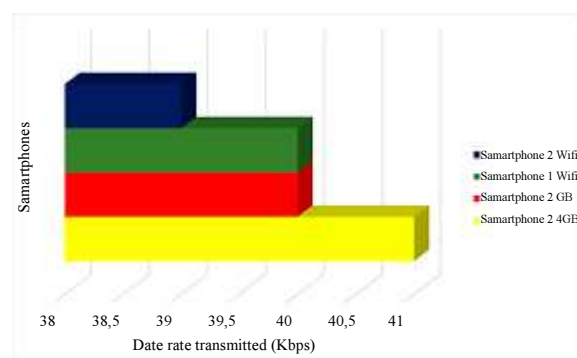
The Fig. 10(b) shows the data rate transmitted. Although the response time of 4G is lower than the wifi network due to the amount of network elements in a wifi network. We can observe that the amount of data required by the 4G network more than the wifi network due a greater range of information transmitted and a low competition for the communication channel. The Wi-Fi has a higher collision and competition rate of the communication channel divided the division of communication with other services and applications of the city, thus reducing the bit rate of transmission.

The Fig. 10(c) shows the percentage of Battery Consumption. We can observe that the greater consumption of the battery use is greater when the data is transmitted through the Wi-Fi networks. This is because the transmission of data in this network consumes more battery due to distance with the antenna and the amount of requests lost by the network due to collisions in the network. Moreover there was a higher consumption of battery due to the amount of retransmissions in the Wi-Fi network divided the amount of messages collisions as well as a competition with the information generated by the city.

Parents Profile

When we analyze the application through the parents' view the amount of transferred data is greater due to the visualization of the map and geolocalization information of this map. Figure 11(a) describes data from the data rate transmitted, in which we can observe that the consumption of the wireless network is greater than in the 4G due to the amount of information relayed on the network. The difference in the amount transmitted among the mobile phone refers to the broadband characteristics of each mobile device. In addition, since the wireless network communication is between the device and the internal access point, that is, the access point installed in the person's home, the competition for the network communication channel is lower due to the low amount of devices connected. This increase in transmission rate impacts on battery consumption.
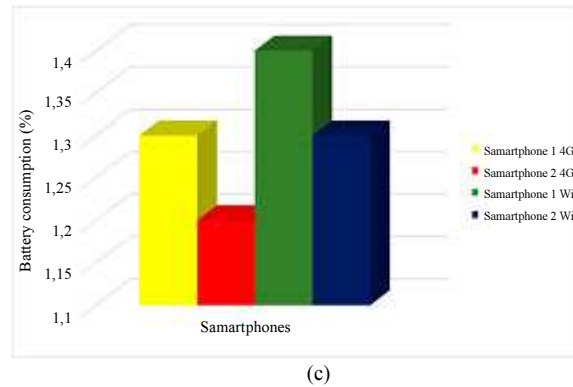

(a)

(b)

(c)

Fig. 10: Results obtained through test, taking into consideration (a) Average time to update the database (b) Data Rate transmitted and (c) Percentage of battery consumption when consider the driver profile
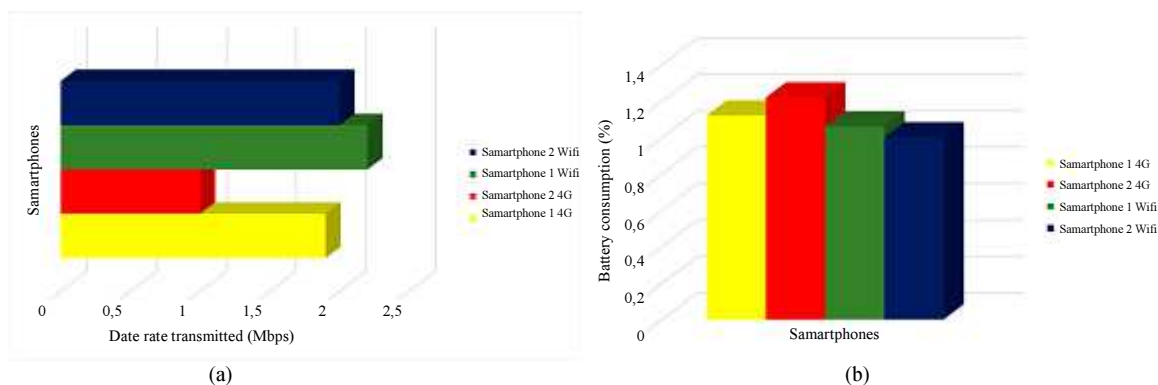

(a)

(b)

Fig. 11: Results obtained through test, taking into consideration: (a) Data Rate Transmitted and (b) Percentage of Battery Consumption when consider the Parents profile

The battery consumption was higher when the data were transmitted through the 4G network because the size of the data transferred was larger than when the application was executed in the driver module. So this packet size made the power consumption larger than the Wifi network as we can observe in Fig. 11(b).

Final Remarks

We can observe that even with a communication infrastructure that is currently precarious, that is, by dividing the communication of the wireless network with other information of management of the city of Catanduva, as well as with the use of 4G technology to transmit the cellular network. The ICCV has a good transmission rate, i.e. a low consumption of the data and communication package for parents and motorcycles, with a low energy consumption.

Such a performance could be improved through the use of 5G, although such technology was not homologated in Brazil and mobile phone manufacturers do not provide such technology in the city. In addition, the use of switches and routers in which it provides a Software-Defined Network (SDN) support could isolate

the flow of information from the ICCS causing competition and the number of packet collisions to decrease, thereby increasing the performance of Wi-Fi.

Conclusion

In the present work, a new complete infrastructure was developed for the abstraction, storage and processing of the information of the sensors that are loaded onto the vehicle, called ICCV. ICCV has an architecture that aims to provide services and information to optimize the transport and access to the information not only of the vehicle, but also of the vehicular traffic on the roads. In addition, a new device for reading sensors embedded in the vehicle, called ORD, has also been developed in order to propagate information not only to a mobile device of the user, but also to establish a direct communication with a control center, reducing the time of dissemination of the information between reader and central. Therefore, streamlining the processing and propagation of this information to different services. In this work we did not consider the sum of information security mechanisms on which it is based to ensure the privacy of parents and

students, as well as the safety of the school van route preventing a GPS injection attack. Thus, such development would remain as future work.

As a future work, it is intended to improve the reading device, in order to minimize the electronic circuit, besides conducting a qualitative and quantitative analysis in relation to the time of communication between the device and the service center, as well as between the user and service response times. Finally, we intend to develop new services with the data collected.

Acknowledgement

Author's Contributions

Daniel Rodrigues, Jessica N. Severino and Andre L. Cristiani: Participated in the development of work, in the evaluation of the results and in the writing of the paper.

Fabiano T.M. Costa, Luis H.V. Nakamura and Rodolfo I. Meneguette: Participated in the coordination of the work, elaboration of the research plan, organization of the studies and in the writing of the paper.

Ethics

The authors agree with the publication of this manuscript, which does not contain ethical issues. All references are stated in the references section.

References

Amarasinghe, M., S. Kottegoda, A.L. Arachchi, S. Muramudalige and H.M.N.D. Bandara *et al*., 2015. Cloud-based driver monitoring and vehicle diagnostic with obd2 telematics. Proceedings of the IEEE International Conference on Electro/Information Technology, May 21-23, IEEE Xplore Press, Dekalb, IL, USA, pp: 505-510. DOI: 10.1109/EIT.2015.7293433

Arada, 2018. Arada system. Acessado em. http://www.aradasystems.com/

Charette, R.N., 2009. This car runs on code. IEEE Spectrum, 46: 3-3.

Electronics, E., 2018. Elm327 v2.2. Acessado em. https://www.elmelectronics. com/products/ics/obd/

Euronews, 2017. Euronews. Acessado em Out. http://pt.euronews.com/2013/09/18/carros-autonomos-em-2020

Farrugia, M., J.P. Azzopardi, E. Xuereb, C. Caruana and M. Farrugia, 2016. The usefulness of diesel vehicle onboard diagnostics (obd) information. Proceedings of the 17th International Conference on Mechatronics-Mechatronika, Dec. 7-9, IEEE Xplore Press, Prague, Czech Republic, pp: 1-5.

Farsi, M., K. Ratcliff and M. Barbosa, 1999. An overview of controller area network. Comput. Control Eng. J., 10: 113-120.

Firebase, 2018. Firebase. Accessed in em. https://firebase.google.com

Hilpert, H., L. Thoroe and M. Schumann, 2011. Real-time data collection for product carbon footprints in transportation processes based on OBD2 and smartphones. Proceedings of the 44th Hawaii International Conference on System Sciences, Jan. 4-7, IEEE Xplore Press, Kauai, HI, USA, pp: 1-10. DOI: 10.1109/HICSS.2011.356

Jan, M.A., P. Nanda, X. He and R.P. Liu, 2014. Pascccc: Priority-based application-specific congestion control clustering protocol. Comput. Netw., 74: 92-102.

Li, F., H. Zhang, H. Che and X. Qiu, 2016. Dangerous driving behavior detection using smartphone sensors. Proceedings of the IEEE 19th International Conference on Intelligent Transportation Systems, Nov. 1-4, IEEE Xplore Press, Rio de Janeiro, Brazil, pp: 1902-1907. DOI: 10.1109/ITSC.2016.7795864

Machado, A.S.L. and B.R.R. Oliveira, 2007. O sistema obd (on-board diagnosis).

Marin, M.E., M. Maricaru, F. Constantinescu and A.G. Gheorghe, 2017. Hardware and software approach for teaching automotive networks. Proceedings of the Electric Vehicles International Conference, Oct. 5-6, IEEE, pp: 1-4. DOI: 10.1109/EV.2017.8242097

Meneguette, R.I. and A. Boukerche, 2017. Servites: An efficient search and allocation resource protocol based on v2v communication for vehicular cloud. Computer Networks, 123: 104-118.

Meneguette, R.I., A. Boukerche and De Grande, R. 2016. Smart: An efficient resource search and management scheme for vehicular cloud-connected system. Proceedings of the IEEE Global Communications Conference, Dec. 4-8, IEEE Xplore Press, Washington, DC, USA, pp: 1-6. DOI: 10.1109/GLOCOM.2016.7842271

Pal, A. and M. Pal, 2017. Iot for vehicle simulation system. Int. J. Eng. Sci., 4: 393-393.

Pereira, A., M. Alves and H. Macedo, 2016. Vehicle driving analysis in regards to fuel consumption using fuzzy logic and obd-ii devices. Proceedings of the 8th Euro American Conference on Telematics and Information Systems, Apr. 28-29, IEEE Xplore Press, Cartagena, Colombia, pp: 1-4. DOI: 10.1109/EATIS.2016.7520160

Society of Automotive Engineers, S., 1997. SAE Fatigue Design Handbook. 3rd Edn., Society of Automotive Engineers, ISBN-10: 1560919175, pp: 470.

Solutions, O., 2018. Obd. Acessado em. http://www.obdsol.com/solutions/ development-tools/development-board/

Tuohy, S., M. Glavin, C. Hughes, E. Jones and M. Trivedi *et al*., 2015. Intra-vehicle networks: A review. IEEE Transact. Intel. Transportat. Syst., 16: 534-545.

Wang, Q., J. Wan and Y. Yuan, 2018. Locality constraint distance metric learning for traffic congestion detection. Pattern Recognit., 75: 272-281.

Wei, L. and D. Hong-ying, 2016. Real-time road congestion detection based on image texture analysis. Procedia Eng., 137: 196-201.

Xie, Y., X. Su, Y. He, X. Chen and G. Cai *et al*., 2017. Stm32-based vehicle data acquisition system for internet-of-vehicles. Proceedings of the IEEE/ACIS 16th International Conference on Computer and Information Science, May 24-26, IEEE Xplore Press, Wuhan, China, pp: 895-898. DOI: 10.1109/ICIS.2017.7960119

Zhang, M., C. Chen, T. Wo, T. Xie and M.Z.A. Bhuiyan *et al*., 2017. Safedrive: Online driving anomaly detection from large-scale vehicle data. IEEE Trans. Industrial Informatics, 13: 2087-2096.