Research Article

Proof of Concept of an IoT-Based Public Vehicle Tracking System, Using LoRa (Long Range) and Intelligent Transportation System (ITS) Services

Ricardo Salazar-Cabrera [iD],[1] **Álvaro Pachón de la Cruz** [iD],[2] **and Juan Manuel Madrid Molina** [iD][2]

[1]*Telematics Department, Universidad del Cauca, Popayán, Colombia*
[2]*ICT Department, Universidad ICESI, Cali, Colombia*

Correspondence should be addressed to Ricardo Salazar-Cabrera; ricardosalazarc@unicauca.edu.co

Systems trying to solve mobility issues in cities, such as high levels of accidents and traffic congestion, have been developed worldwide. Intelligent Transportation Systems (ITS) services focused on urban public transport are an option contributing to solve such issues. A few intermediate cities in the Latin American context have developed some of these ITS services, which are mainly based on a tracking system for public transport vehicles. Such tracking systems have great limitations in terms of coverage, availability, and operational cost. In addition, they are commonly isolated mobility solutions, which cannot be easily integrated with other mobility services in the city because they are not based on an ITS architecture. In order to improve public transportation systems in intermediate cities, we proposed the development of an IoT-based public vehicle tracking system, using LoRa (Long Range) and ITS services. In this research, we developed the proposed system as a proof of concept. We designed and executed some experiments, in order to adjust parameters of the LoRa technology and to test its operation. This article presents the methods we followed for developing the proof-of-concept model, a description of the experiments, and their results. The results lead to conclude that the LoRa technology and an IoT-based system are adequate for implementation of a mobility service such as the one we propose, once important technical restrictions related mainly to Line of Sight (LoS) are considered. Key aspects for implementation were also identified for deploying the service (as a prototype) in the city of *Popayán*.

1. Introduction

Traffic accidents are one of the leading causes of deaths worldwide [1]. Colombia has a high percentage of deaths (between 27 and 28%), due to traffic accidents in recent years [2]. This problem is serious in intermediate Colombian cities (e.g., *Neiva, Pereira,* and *Popayán*), accounting for 36% of deaths. In addition, cities in the Latin American context have remarkable issues with regard to traffic congestion. This is evidenced in indicators such as the TomTom Traffic Index [3].

In Colombia, 25% of passengers suffering injuries in traffic accidents are passengers of public transport vehicles [4]. Such data are quite worrisome if we consider the proportion of public vehicles in Colombia. Similar situations have been identified in other Latin American countries such as Perú, Chile, México, and Ecuador [5–8]. The most common public transport vehicles in intermediate cities in Colombia and other developing countries are buses and microbuses, with a capacity ranging from 12 to 20 passengers [9]. These vehicles share the road with all the other vehicles, increasing the risk of traffic accidents. Thus, it is important to establish improvement actions to control public transport vehicles in intermediate cities without public transit exclusive lanes. The aim is to reduce the death rates due to traffic accidents and to contribute to lessen traffic congestion in these cities.

A wide variety of ITS (Intelligent Transportation System) services seeking to improve public transport service

conditions have been deployed worldwide. The vast majority of these systems use public transport vehicle tracking systems. However, there are very few cases in which the deployed ITS services use an ITS architecture specifically defined for the city. Therefore, integration of mobility services in the city is considerably difficult. In addition, only a few recent cases use improved communication technologies because the vast majority uses technologies based on mobile telephony, which has major deficiencies in coverage, availability, and operating cost.

Some of the most salient work identified in our literature review is presented below. In works such as those by Bojan et al. [10], Zambada et al. [11], Mukheja et al. [12], and Azer and Elshafee [13], ITS systems were developed to track vehicles using cellular communication technologies such as GSM and GPRS, which have great technical limitations and high operating costs. Additionally, these proposals do not use an ITS architecture as a basis. Other works such as those by Tanaka et al. [14], Boshita et al. [15], and Hattarge et al. [16] introduce the use of LoRa technology and/or its LoRaWAN (Long Range Wide Area Network) communications protocol, to track vehicles; however, such proposals were not based on an ITS architecture, they were not focused on the context of Latin American intermediate cities, and the results obtained with the use of LoRa technology were not presented in sufficient detail.

Considering the aforementioned issues, our proposal is based on an ITS architecture and technologies such as the Internet of Things (IoT) and LoRa (Long Range). Additionally, our proposal details system development, arriving at a proof-of-concept model and validating its operation through some experiments. Our research is carried out in the following four stages: design of the proposed service, development of the proof-of-concept model, execution of experiments, and analysis of results. The proposed service design is summarized because it is detailed in a previous article by the authors [17].

The development of the system focuses mainly on the components considered as critical, related to communication between vehicles and the management center and also with presentation of information to users of the system. Communication between vehicles and the management center requires sending information through gateways. Such gateways receive vehicle information using LoRa technology and transmit it to the management center using the Internet. The ideal parameters for the use of LoRa technology were not clear for the proposed mobility service, so it was necessary to develop experiments during the development of the system and then to validate its operation. Considering the context, budget limitations were also taken into account for developing the system, by using affordable hardware and free software tools. The executed experiments allowed us to determine aspects of the operation such as range of LoRa transmission, percentage of received packets, and number of messages per minute. The number of messages was determined considering the size of the message and the maximum number of vehicles that can be managed from each gateway. Finally, the analysis of results shows important restrictions that must be taken into account in the system development, mainly related to the operation with LoRa.

The remainder of the article is organized as follows: there are sections on materials and methods, executed experiments, obtained results and discussion, and conclusions and future work.

2. Materials and Methods

The research was carried out in four stages: design of the proposed service, development of a proof-of-concept model, execution of experiments, and analysis of results. The four stages are presented below in detail.

2.1. Design of the Proposed Service. A previous work of the authors features the design of an IoT-based public vehicle tracking system, using LoRa technology and ITS architecture [17]. This design was made in the five steps presented below.

In the first step, we defined a suitable ITS architecture for intermediate cities in the Latin American context. The ITS architecture deemed as adequate was based on the American ITS architecture known as ARC-IT [18].

In the second step, we created the *detailed service diagram*, using the defined ITS architecture and the service description diagrams used by ARC-IT, for services PT01 and PT08 (PT01 is Transit Vehicle Tracking and PT08 is Transit Traveler Information in ARC-IT). The detailed service diagram is presented in Figure 1. This figure shows the physical objects involved in service provision, the actors (traveler and system operator), and the information messages exchanged between actors and objects, and between physical objects.

In the third step, we selected the appropriate communication technology to link the Transit Management Center and the OBE (On-Board Equipment) on the transit vehicles, which was considered critical. Communication between the gateways and the cloud (where the Transit Management Center would be) uses the Internet. We evaluated LPWA (Low Power WAN) technologies such as LoRa, Sigfox, NB-IoT (Narrow Band IoT), and V2X technologies (Vehicle to Everything) technologies such as DSRC (Dedicated Short Range Communication) and C-V2X (Cellular V2X). After a technical analysis involving several parameters, we deemed LoRa as an appropriate technology for the PTVTS.

In the fourth step, we created the *network diagram*, which features the remaining technological components of the system. The network diagram is presented in Figure 2. The device (OBEs and Traveler Support Equipment), communication (LoRa, WiFi, and Internet), and application (web apps and software in the cloud Center subsystem) layers are clearly identified in the diagram. These layers are commonly used in IoT service architectures, which makes the designed service an "IoT-based ITS." We determined that intermediate stations with a LoRa gateway are required on the streets the vehicles use; such gateways could be located at some bus stops. These intermediate stations are necessary to collect the Transit Vehicle OBE's information.

Finally, in the fifth step, we performed an analysis of the service required characteristics for the city of *Popayán* (Colombian intermediate city), by defining the number of required intermediate stations, the number of components,
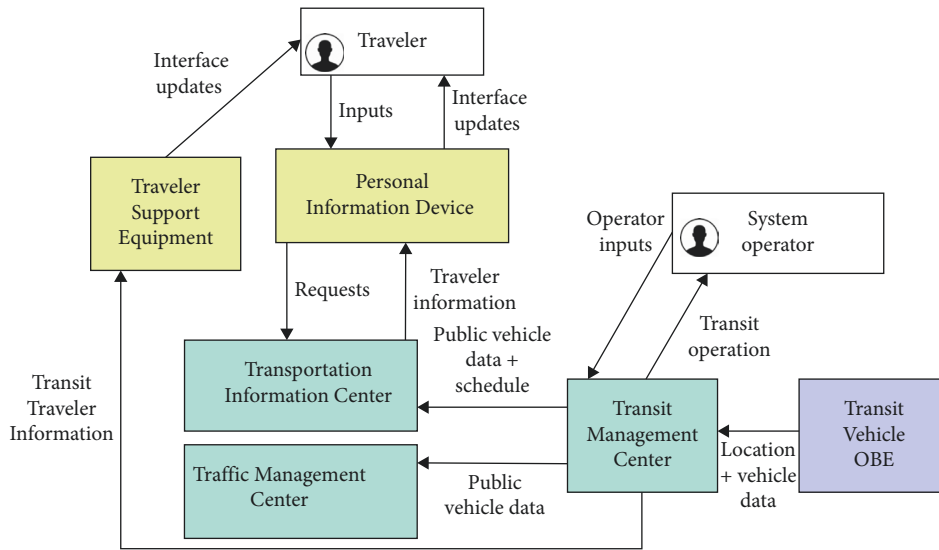
FIGURE 1: Detailed service diagram of public transport vehicle tracking service (PTVTS).
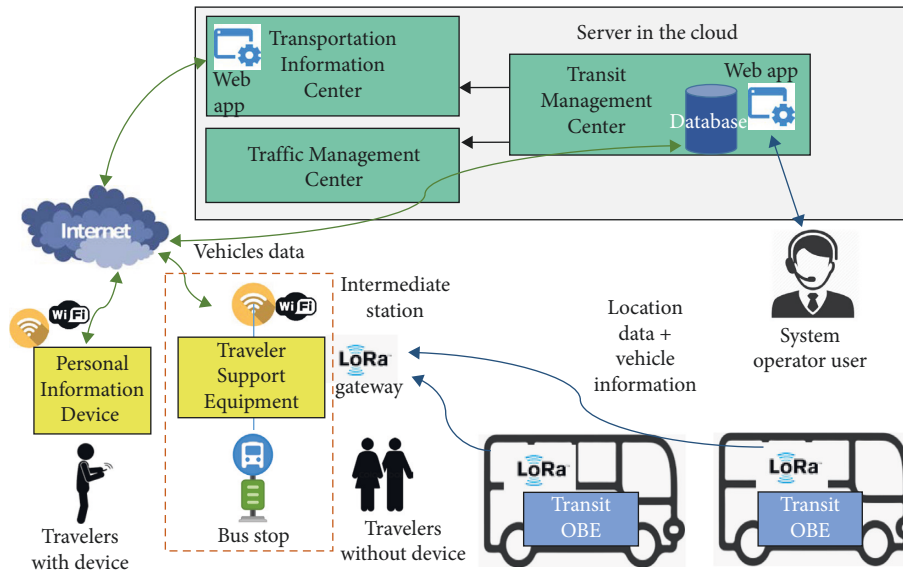

FIGURE 2: Network diagram of the public transport vehicle tracking service.

the recommended references for the necessary elements, an implementation budget, and an annual maintenance and support budget.

2.2. Development of a Proof-of-Concept Model. The development of the system focused mainly on the components considered as critical, for communication between the vehicles (Transit Vehicle OBE) and the server in the cloud (the *Transit Management Center* component), passing through the LoRa gateway equipment. Priority was given to these components because they are responsible for obtaining the field information and sending it to the server in the cloud. The *Personal Information Device* component was included in the development, mainly to show users system information through an appropriate interface, such as a web page.

Figure 3 presents the network diagram for the proof-of-concept model.

The additional components of the designed service (Transportation Information Center, Traffic Management Center, and Traveler Support Equipment) were not developed for this version of the system because it was advisable to first assess the results obtained with the critical components, to evaluate the viability of our proposal.

For the development of the system, we first purchased the elements in the OBE and in the bus stops. Due to project budget restrictions, only two microcontroller cards and a gateway device were purchased. The selected microcontroller card was the HiLetgo ESP32 LoRa 0.96 inch OLED display, manufactured by Heltec (see Figure 4). The selected GPS was the Ublox Neo 6M. Regarding the gateway, we had the opportunity to review two references, the
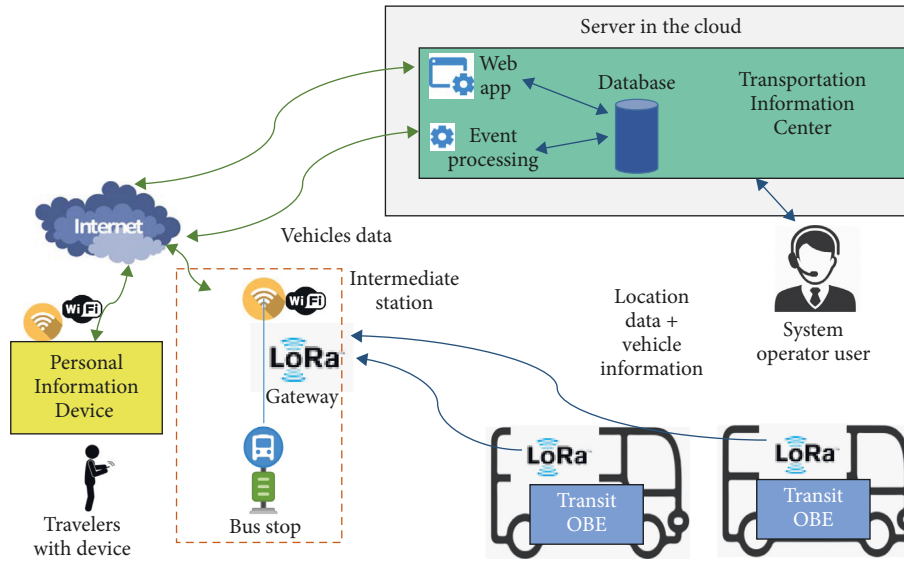
Figure 3: Network diagram of the proof-of-concept model.

Figure 4: Selected microcontroller card, HiLetgo ESP32 LoRa 0.96 inch OLED display (Heltec).

Figure 5: Selected LoRa gateway, Dragino LG01.

RAKWireless RAK831LoRa/LoRaWAN and the Dragino gateway LG01. After performing some tests with the two gateways, we deemed the Dragino gateway LG01 as the best option (see Figure 5) because this reference did not require an additional card (such as a Raspberry Pi) to operate. In addition, the Dragino gateway was easily programmable through the Arduino IDE, which was not possible with the other gateway.

Once we selected and acquired the elements, we carried out some experiments prior to final development (see the *Executed Experiments* section). Finally, we developed the software required to send the information collected by the microcontroller card (main OBE component of each vehicle) to the gateway, through LoRa messages. The information for this case was vehicle location (latitude and longitude), speed, and altitude, taken from the GPS device. We also developed the software for receiving the LoRa messages sent by the OBEs and transmitting the information through the Internet to the server in the cloud. Both modules were developed

using the Arduino IDE (Integrated Development Environment) tool. The source code for the modules is available online (see the *Data Availability* section).

In order to send the information from the OBE devices to the gateway, we evaluated the possibilities (through testing) of establishing a LoRaWAN or simply sending LoRa messages. The LoRaWAN specification is a Low Power, Wide Area (LPWA) networking protocol designed to wirelessly connect battery operated "things" to the Internet in regional, national, or global networks [19]. LoRaWAN architecture is deployed in a star-of-stars topology, in which gateways relay messages between end devices and a central network server. The gateways are connected to the network server via standard IP connections and act as a transparent bridge, simply converting RF packets to IP packets, and vice versa [19]. We used the TTN platform (The Things Network) to perform tests with LoRaWAN. The Things Network (TTN) enables low-power devices to use long-range gateways to connect to an open-source, decentralized network to exchange data with applications. The connection of devices to TTN could be made by OTAA (over-the-air activation) mode or ABP (activation by personalization) mode. OTAA is the preferred and most secure way to connect with TTN [20]. In OTAA, when devices join the network, a dynamic

DevAddr (Device Address) is assigned and encryption keys are negotiated with the device. ABP mode is used when it is required to encrypt the DevAddr and device encryption keys [20]. We used LoRaWAN in OTAA mode, but the connection time of the devices was too long, compared to the sending time of LoRa messages without the LoRaWAN protocol. This might be due to some particularity of the LoRaWAN protocol, or the processing done in the test. We reviewed literature in this regard [21–23] trying to find a solution and ran some tests related to data rate and access channels (detailed in experiment 2). Finally, we achieved connection to the LoRaWAN in an acceptable time of a few seconds; however, connection through LoRa messages proved to be faster. Thus, considering vehicles connect to several gateways along their route, we decided to send LoRa messages as the best option, without establishing a LoRaWAN.

The sending and receiving of LoRa messages require some special parameters to be determined, such as the frequency, the Spread Factor (SF), and the bandwidth (BW). Regarding the frequency, we worked in the 902–928 MHz range, used in the United States (US), selecting 915 MHz as the channel for sending/receiving data. According to the LoRa Alliance in its Regional Parameters document v1.1. [24], this would be the frequency range allocated to Colombia for this technology. Literature reviewed [25–28] on LoRa recommends some values for SF and BW; however, we performed some tests modifying those values (experiment 2, *Executed Experiments* section), with the aim of optimizing range, message size, and frequency of sending messages. We determined that the appropriate values were 7 for the SF (which can be set between 1 and 12) and 125 KHz for the BW (which can be 125 KHz, 250 KHz, or 500 KHz). Another important parameter is the *Sync Word*, which must be the same for both sender and receiver.

In addition to the software for the OBE and the gateway, we wrote a program in the cloud server, for receiving the messages from the gateway and storing them in the *Event Processing* database (Figure 3). We used the XAMPP platform ([29]), which features a web server (Apache), a programming tool (PHP), a database engine (MariaDB), and a database administrator (phpMyAdmin). We used RESTful actions for communication between the LoRa gateway and the *Transit Management Center* in the cloud. These actions use the POST HTTP method in the LoRa gateway software to send information of all vehicles and the GET HTTP method in the cloud server program to capture the data and record it in the database. A link to the source code for this program is also included in the *Data Availability* section of this article.

Finally, we developed a web application using XAMPP, for the two types of users identified for the system (operator and traveler), so that they could visualize the location of the public transportation vehicles on a map. Figure 6 shows the user login interface. Figure 7 shows the operator user interface, for querying the location of any vehicle at any time and date of the day. Results are presented in a table and also on a map, using Google APIs. The end user can also see the

FIGURE 6: Login interface for the system user.

latest location of a vehicle, but the query is limited to current date/time and to a certain number of last locations.

2.3. Executed Experiments. We executed three experiments to test the operation of the components and their integration. Two additional experiments were also designed and executed to evaluate system operation.

The first experiment, carried out during the development of the system, was to establish a point-to-point connection between the two microcontroller cards for sending a test LoRa message. One of the cards was programmed as sender and the other as receiver. Several test messages were sent every 3 seconds from the card configured as sender, expecting to receive all the messages on the card configured as receiver without any problems. To view the messages sent and received, we used the *Serial Monitor* in the Arduino IDE. Figure 8 presents the architecture for experiment 1.

In the second experiment, we used the same configuration of the initial experiment, but the card acting as a LoRa receiver was also configured as a WiFi connection client (using the WiFi internal module in the ESP32 card) in order to relay the received information to a web server located on a computer in the same WiFi network (using XAMPP and a program named *Event Processing*). The *Event Processing* module received the messages and stored them in a local database. In this experiment, we aimed at determining the appropriate SF and BW parameters, by varying them and verifying in which cases a signal with greater power was obtained in the receiver, using the parameter received signal strength indicator (RSSI), measured in dBm. Several test messages were also sent every 3 seconds from the card configured as sender. We used the *Serial Monitor* tool of the Arduino IDE to visualize sent and received messages. We used XAMPP's phpMyAdmin tool to visualize the data recorded in the database. Figure 9 presents an architecture for experiment 2.

In the third experiment, the Dragino gateway was configured to receive the test LoRa messages from the two microcontroller cards, both configured as sender only. The Dragino gateway was also configured to send the information to the web server used in the previous experiment. In this experiment, tests were carried out to determine an adequate message size, to be able to send them with a few seconds of

Vehicle Location - For Operator

OF VEHICLE: 2 ▾ DATE: 25/06/2019   HOUR: 03:43:56 p. m.   MAX OF LOCATIONS: 5   [Enviar]

LAST LOCATIONS OF VEHICLE IN SELECTED DATE, BEFORE SELECTED HOUR, MINUTES AND SECONDS

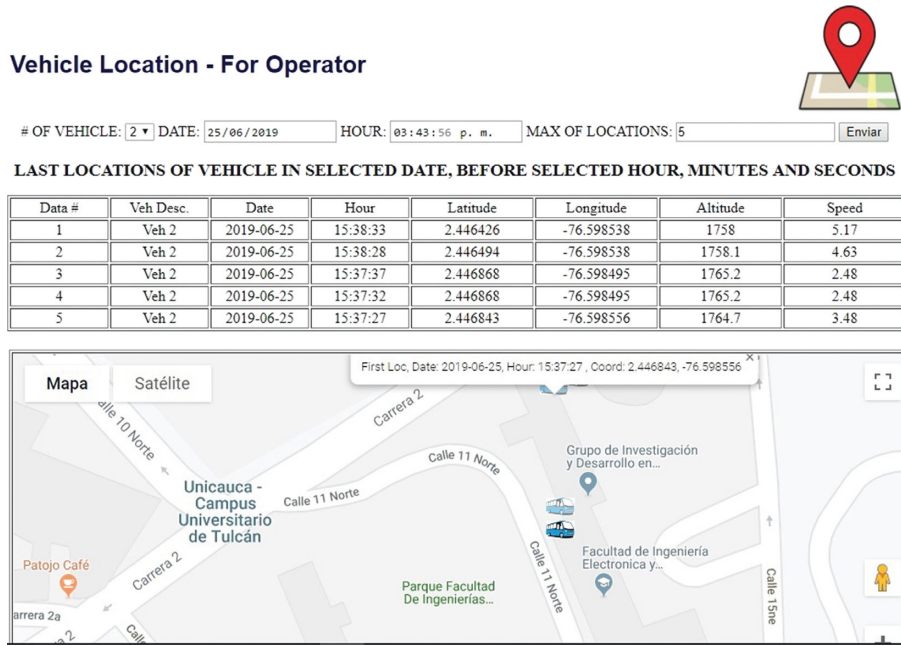| Data #	Veh Desc.	Date	Hour	Latitude	Longitude	Altitude	Speed
1	Veh 2	2019-06-25	15:38:33	2.446426	-76.598538	1758	5.17
2	Veh 2	2019-06-25	15:38:28	2.446494	-76.598538	1758.1	4.63
3	Veh 2	2019-06-25	15:37:37	2.446868	-76.598495	1765.2	2.48
4	Veh 2	2019-06-25	15:37:32	2.446868	-76.598495	1765.2	2.48
5	Veh 2	2019-06-25	15:37:27	2.446843	-76.598556	1764.7	3.48

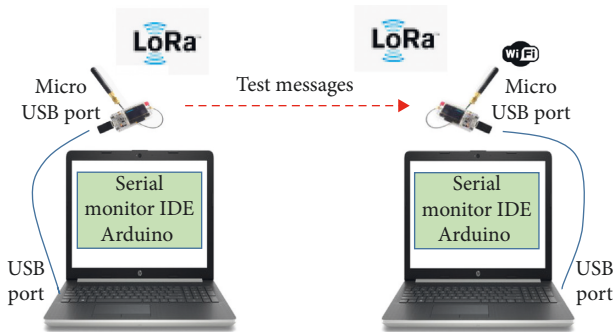Figure 7: Public vehicle tracking interface for the system operator.
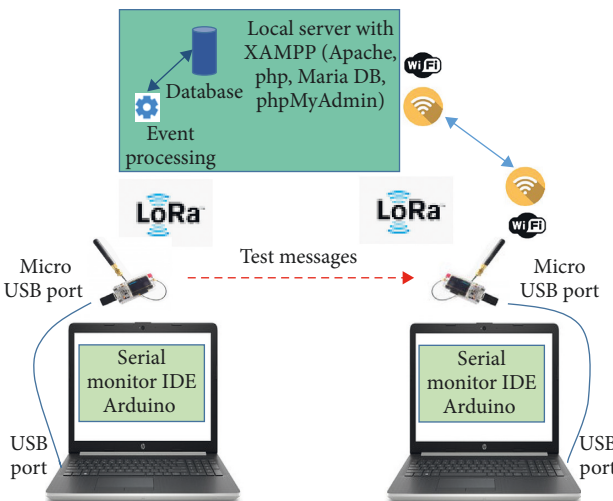
Figure 8: Experiment 1 architecture.

Figure 9: Experiment 2 architecture.

difference and to receive messages from the two cards at the gateway. This is due to the fact that mobility services related to tracking require sending messages with a high frequency (for example, 10 times per minute), so that the location can be determined in real time or as close as possible.

In this experiment (third), we performed tests with LoRaWAN in the OTAA mode, showing that the devices took much time registering on the network. The cause for this long connection delay was researched [21–23]. We found out that a LoRaWAN operates in several frequency channels that are determined by the gateway configuration. The number of allocated channels depends on regional restrictions and the network options [21]. We also found out that the data rate parameter defined by the SF and the bandwidth is relevant to the LoRaWAN configuration. Some tests related to these mentioned aspects (access channels and data rate) were performed, initially modifying the access channels and maintaining the data rate, and subsequently improving at this point and identifying the ideal channels, the data rate was modified, identifying the best channel, which allowed to reduce the connection time of the devices to the network to a few seconds. However, the connection time through LoRa messages, without the establishment of a LoRaWAN, remained faster and more efficient. In addition, the type of service to be implemented requires that the devices (located in the vehicles) connect to several gateways along their route, so we consider sending LoRa messages as ideal, without using the LoRaWAN configuration. Figure 10 presents the architecture for experiment 3.

The fourth experiment was executed with the system already developed. The microcontroller cards were programmed with the latest version of our software, which includes the correct GPS configuration, the appropriate size of the message, the LoRa parameters (SF, BW, and Sync Word). The software was configured to only send information without receive acknowledge, in order to be able to send messages more frequently. The Dragino gateway software was also configured with the latest version of our software, including LoRa parameters, Internet connection
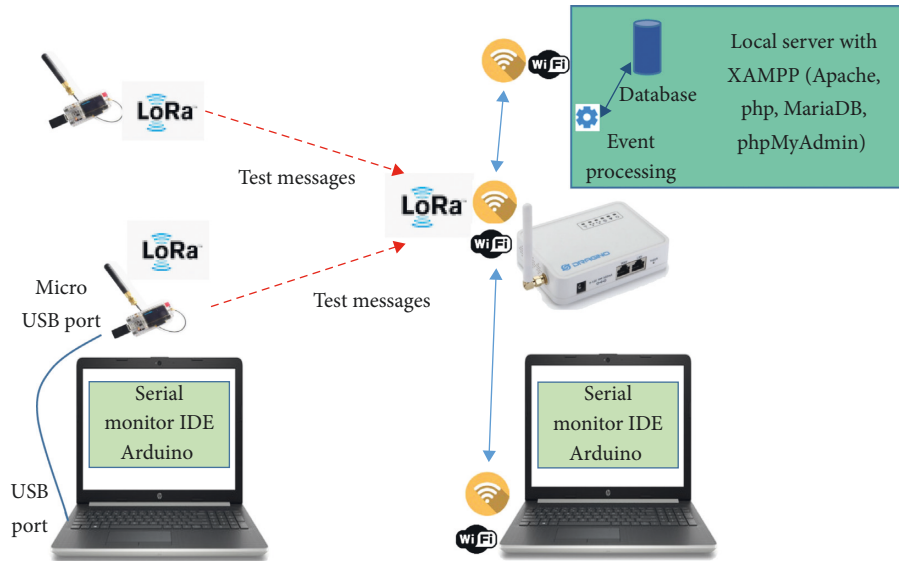
Figure 10: Experiment 3 architecture.

via WiFi, and sending data to a server in the cloud. The gateway was located on the terrace of a 4-floor building of the University of Cauca, and the OBE devices were located in the University Sports Center, which is in front of the aforementioned building, at approximately 400 meters. Information collected by each one of the microcontroller cards was sent every 5 seconds approximately; we expected to receive all sent messages in the Dragino gateway. Figure 11 presents the architecture for experiment 4.

Finally, in the fifth and last experiment, we used the same setup of the fourth experiment, but the two OBE devices were located in vehicles, and each vehicle did a different route through the streets near the building where the gateway was located. The route for each vehicle was approximately 2 km. The size of the messages and the sending frequency was the same as in the previous experiment. Experiment 5 architecture is similar to Figure 11, but the OBE devices were located in vehicles.

3. Results and Discussion

We present the results obtained in the three experiments performed during system deployment and two experiments performed with the already deployed system.

In the first experiment, the communication of the two LoRa modules (end devices) was tested at varying distances (from 3 to 200 meters), with the first module acting as sender and the other as receiver. We verified the correct sending and reception of data.

The results for the first experiment were successful. 100% of the sent packets were received, even with some minor obstacles (walls and windows) in the LoS. When the number of obstacles in the LoS increases, messages were not received. We increased the frequency of sending messages to determine if this affected their correct reception; however, packets were not lost even sending a message every 3 seconds, a rate much higher than the required rate for the deployed service.

In the second experiment, we tested the two LoRa modules with the same configuration of experiment 1. In addition, the receiver LoRa module was also configured as a gateway to send the data to a local server.

The results for the second experiment were successful too. 100% of the messages sent were received, as long as there were no significant obstacles in the LoS. We performed tests with distances up to 400 meters, sending messages once every 3 seconds. During this experiment, we measured the RSSI parameter, to determine the signal levels in the receiver by varying distances between modules and adding obstacles. When we only varied the distance, differences in RSSI levels of the messages were not considerable. When we increased the number of obstacles and the distance, the RSSI levels decreased considerably in the cases where the message was received; in some cases (with a high number of obstacles), the messages were not received. Additionally, in this experiment, maintaining a fixed distance and varying the number of obstacles, we modified the SF and BW parameters to determine which combination would yield a better RSSI. We determined that ideal values were 7 for SF and 125 KHz for BW.

In the third experiment, we tested the two LoRa modules acting as sender and the Dragino device acting as receiver and gateway. During the sending of data in the third experiment, we identified that although LoS existed between the end device and the gateway, some of the sent packets were not received or were received with unexpected characters in the middle of the data. When this happened, the Dragino gateway stopped receiving data (specifically from the device from which the wrong data were received) for a few seconds, and then the gateway resumed receiving data normally from the two devices. In the tests carried out, approximately 2% of the packets sent from each of the cards acting as senders were not received. We determined that reception problems are due to interference between the two signals that arrive at the gateway; however, taking into
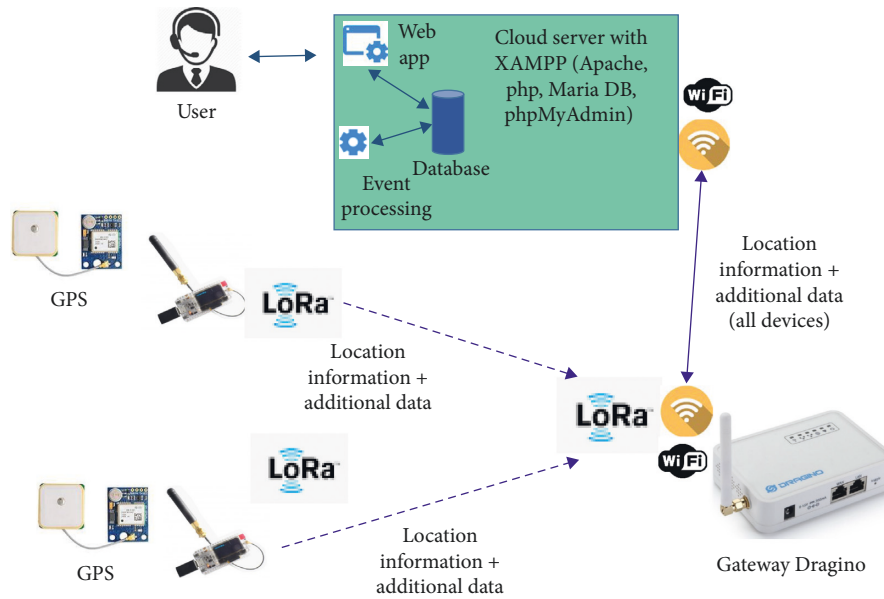
FIGURE 11: Experiment 4 architecture.

account the frequency with which the packets are sent (1 time every 3 seconds), this problem does not represent a major inconvenience for tracking public transport vehicles in real time. However, it is recommended for future work, to perform some tests with a greater number of end devices, to verify if the percentage of not received packets is increased or maintained. If that 2% is maintained or does not increase too much, detected interference would be manageable.

In the fourth experiment, the components were configured in a similar way to the third experiment, but using real GPS-generated data, and a cloud server for data collection. Experiment 4 showed some packet loss from each of the end devices, approximately 2%, similar to the results of experiment 3. Although we tried to decrease the packet sending frequency, packet loss was not affected. To avoid inserting erroneous data in the database, when the packet arrived with unexpected characters, we made a validation in the *event processing* program that allowed us to identify format errors. Packets with errors were not recorded in the database. In this experiment, we did not find LoS problems.

Finally, in the fifth experiment, the configuration of components was the same as the previous experiment, but with moving end devices. Initially, the experiment was not successful; a large percentage (greater than 60%) of the sent messages was not received. The cause that was initially considered the most likely was the large number of obstacles in the LoS. Although the Dragino gateway was located in the roof of a tall building (approximately 25 meters), it was not possible to obtain LoS with most sections of the routes, and the obstacles were considerable (mainly, other buildings). To improve the percentage of received messages, we researched some other possible causes and solutions [21–23]. We found out that the main cause of the high percentage of lost packets was actually due to a large number of obstacles in the LoS and the Spread Factor used because the value of 7 has a low range. According to [22], increasing the SF parameter (to a value between 8 and 12) also increases the range that can be

achieved but decreases the data rate and increases the ToA (Time on Air, elapsed time since the packet is sent until it reaches its destination). In our experiment, and in general for public vehicle tracking, a high data rate is not required, while a good signal range is preferred.

One possible way to modify the relevant LoRa parameters is through an adaptive data rate (ADR) algorithm [23], which features an automatic learning method using the different transmission parameters. However, considering the low number of devices in our experiment and the low number of parameters to modify, we consider this approach is not necessary in our case.

Considering our findings, we reran experiment 5 and measured the new results. We decided to manually modify the SF parameter from 8 to 12, making measurements of the lost packets for each of the values, keeping the BW parameter fixed at 125 KHz. When performing tests with SF values greater than 7 until reaching a value of 12, the best results were obtained with an SF of 10, achieving a decrease in the percentage of lost packets from 60% to 28%, with a final data rate of 10 messages per minute. Additionally, it is important to mention that the largest number of lost packets was presented at sites with a large number of obstacles in the LoS.

The most salient results of our experiments are as follows:

(i) LoS between the gateway and the end devices in our system was very relevant, due to the fact that the number of possible obstacles to establish communication was relatively low.

(ii) The change in the size of the message (between 10 and 35 characters) was not relevant to improve the communication conditions, considering that we only had two devices communicating with a gateway, sending one message at least every three seconds. However, according to the reviewed literature, when the number of devices that communicate with

the gateway is high, having a certain message size could affect the maximum possible message frequency.

(iii) The *Bandwidth* and *Spread Factor* parameters recommended for the frequency range used in the US (902–928 MHz, same for Colombia) are 125 KHz for bandwidth and a Spread Factor of 7. However, we verified that if a high data rate is not required, it is possible to use Spread Factors greater than 7 (between 8 and 12), to reduce the number of lost packets.

(iv) When we tested two devices sending LoRa messages to the gateway, approximately 2% of the total packets sent were lost. We consider that such packet loss may be due to interference between the two signals; however, because the frequency of sending messages is relatively high, this loss would not be significant to track public vehicles.

(v) Although the number of lost packets in the final experiment (where moving vehicles were used) is relatively high (28%), we found that lost packets occurred mainly in sectors where there were too many obstacles in the LoS. This calls for the importance of locating the gateways in places tall enough to avoid as many obstacles in the LoS as possible.

4. Conclusions and Future Work

The proposed tracking system for public transport vehicles through an IoT-based system, LoRa technology, and ITS services is suitable for implementation in an intermediate city, if some important technical aspects of LoRa operation are considered. These technical aspects were identified in the experiments carried out in this work.

The most relevant technical aspects of LoRa operation to be taken into account are an adequate LoS between the gateway and the devices in the vehicles, which requires a strategic location of the gateways in places with sufficient height and visibility to the roads from which the vehicles will send the messages (number of obstacles between the gateway and the roads should be as small as possible); the Spread Factor (SF) should be adjusted between 8 and 12 for each one of the gateways, and tests for packet loss should be performed in the area where the messages are expected to be received. In case it is not possible to carry out the tests, an SF of 10 is recommended, according to the tests performed in our work; for the recommended SF to be a good option, data rate should not be too high because the ToA is increased.

With regard to the technical characteristics of the developed system, it was possible to send messages with location data (latitude and longitude) and speed of vehicles with LoRa technology, with an adequate message size, an acceptable range, a sufficient sending frequency for the type of service, and a manageable packet loss. However, this was achieved in cases where LoS had no obstacles or the number of obstacles was low. Although in the final experiment, approximately a quarter of the messages sent by the devices were not received, in a type of tracking system where the device located in the vehicle sends information with a

sufficiently high frequency (approx. 10 times per minute), such losses are acceptable. In addition, a better location of gateway devices improves LoS and would reduce packet losses considerably.

In our prior research, we had estimated a total number of 15 gateways for covering the city of *Popayán*, which according to the results obtained from the range and LoS experiments is a low number. In addition to the factors that were taken into account in the service design proposal, such as the city dimensions and road location, it is very important to determine strategic tall sites where gateways can be located. In most cases, these gateways will not be able to be located at the bus stops (as estimated in the preliminary design) because range would be very limited.

As a future work, in addition to the strategic location of the system gateways, in order to achieve the widest possible range in communication and avoid higher deployment costs, other relevant parameters could be adjusted in the LoRa communication, e.g., using a larger range antenna in the gateway, or the use of a different brand/model of gateway. Also, for the future, it is very important to test the operation of the system with a greater number of end devices, to determine if this considerably increases packet loss beyond 2% of the total.

Data Availability

The programs data (developed programs for the end devices, the Dragino gateway, and the Event Processing component) used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] World Health Organization, "Report on the world situation of road safety 2015," 2019, http://www.who.int/violence_injury_prevention/road_safety_status/.

[2] National Institute of Legal Medicine and Forensic Sciences, Centro de referencia nacional sobre violencia 2018, 2019, http://www.medicinalegal.gov.co/documents/20143/217010/Boletin+mensual+junio-2018.pdf/57a48178-ba32-2725-caf2-27b9f2079114.

[3] Tom Tom Traffic Index. 2019 https://www.tomtom.com/en_gb/traffic-index/.

[4] D. Vargas, *Comportamiento de Muertes y Lesiones en los Accidentes de Tránsito, Colombia*, National Institute of Legal Medicine and Forensic Sciences, Bogota, Colombia, 2015, http://www.medicinalegal.gov.co/documents/20143/49523/Accidentes+de+transporte+primera+parte.pdf.

[5] INEI (National Institute of Statistics and Informatics of Perú), "Análisis de los accidentes de tránsito ocurridos en el 2014," 2014, https://www.inei.gob.pe/media/MenuRecursivo/publicaciones_digitales/Est/Lib1308/cap03.pdf.

[6] SCT (Secretary of Communications and Transportation, Mexico), "Estadística de accidentes de tránsito," 2017, http://www.sct.gob.mx/fileadmin/DireccionesGrales/DGST/Estadistica_de_accidentes/A%C3%B1o_2017/RESUMEN_2017.pdf.

[7] CONASET (National Traffic Safety Commission), "Estadísticas generales de accidentes de tránsito, Chile," 2017, https://www.conaset.cl/programa/observatorio-datos-estadistica/biblioteca-observatorio/estadisticas-generales/.

[8] National Institute of Statistics and Census, "Anuario de transporte 2016, ecuador en cifra," 2016, http://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Economicas/Estadistica%20de%20Transporte/2016/2016_AnuarioTransportes_%20Principales%20Resultados.pdf.

[9] S. Valbuena, *Muertes y Lesiones No Fatales en Accidentes de Tránsito, Colombia*, National Institute of Legal Medicine and Forensic Sciences, Bogota, Colombia, 2011, http://www.medicinalegal.gov.co/documents/20143/49511/Accidentes+De+Transito.pdf.

[10] T. M. Bojan, U. R. Kumar, and V. M. Bojan, "An internet of things based intelligent transportation system," in *Proceedings of the 2014 IEEE International Conference on Vehicular Electronics and Safety*, pp. 174–179, IEEE, Hyderabad, India, December 2014.

[11] J. Zambada, R. Quintero, R. Isijara et al., "An IoT based scholar bus monitoring system," in *Proceedings of the 2015 IEEE First International Smart Cities Conference (ISC2)*, pp. 1–6, IEEE, Guadalajara, Mexico, October 2015.

[12] P. Mukheja, M. K. Kiran, N. R. Velaga, and R. B. Sharmila, "Smartphone-based crowdsourcing for position estimation of public transport vehicles," *IET Intelligent Transport Systems*, vol. 11, no. 9, pp. 588–595, 2017.

[13] M. A. Azer and A. Elshafee, "A real-time social network-based traffic monitoring & vehicle tracking system," in *Proceedings of the 2018 13th International Conference on Computer Engineering and Systems (ICCES)*, pp. 163–168, IEEE, Cairo, Egypt, December 2018.

[14] M. S. Tanaka, Y. Miyanishi, M. Toyota et al., "A study of bus location system using LoRa: bus location system for community bus "Notty"," in *Proceedings of the 2017 IEEE 6th Global Conference on Consumer Electronics (GCCE)*, pp. 1–4, Nagoya, Japan, October 2017.

[15] T. Boshita, H. Suzuki, and Y. Matsumoto, "IoT-based bus location system using LoRaWAN," in *Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 933–938, IEEE, Maui, HI, USA, November 2018.

[16] S. Hattarge, A. Kekre, and A. Kothari, "LoRaWAN based GPS tracking of city-buses for smart public transport system," in *Proceedings of the 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pp. 265–269, IEEE, Jalandhar, India, December 2018.

[17] R. Salazar-Cabrera, Á. P. de La Cruz, and J. M. Madrid Molina, "Fleet management and control system from intelligent transportation systems perspective," in *Proceedings of the 2019 2nd Latin American Conference on Intelligent Transportation Systems (ITS LATAM)*, pp. 1–7, IEEE, Bogota, Colombia, 2019.

[18] ARC-IT, "National ITS architecture," 2018, http://local.iteris.com/arc-it/.

[19] LoRa Alliance, "What is the LoRaWAN® specification?," 2019, https://lora-alliance.org/about-lorawan.

[20] The Things Network (TTN), "LoRaWAN network server," 2019, https://www.thethingsnetwork.org/docs/network/.

[21] D. Bankov, E. Khorov, and A. Lyakhov, "On the limits of LoRaWAN channel access," in *Proceedings of the 2016 International Conference on Engineering and Telecommunication (EnT)*, pp. 10–14, Moscow, Russia, November 2016.

[22] R. Sanchez-Iborra, J. Sanchez-Gomez, J. Ballesta-Viñas et al., "Performance evaluation of LoRa considering scenario conditions," *Sensors*, vol. 18, p. 772, 2018.

[23] N. Benkahla, H. Tounsi, Y. Song, and M. Frikha, "Enhanced ADR for LoRaWAN networks with mobility," in *Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 1–6, Tangier, Morocco, 2019.

[24] LoRa Alliance, "Regional parameters," 2017, https://lora-alliance.org/sites/default/files/2018-04/lorawantm_regional_parameters_v1.1rb_-_final.pdf.

[25] R. El Chall, S. Lahoud, and M. El Helou, "LoRaWAN network: LoRaWAN network: radio propagation models and performance evaluation in various environments in Lebanon," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2366–2378, 2019.

[26] A. M. Yousuf, E. M. Rochester, B. Ousat et al., "Throughput, coverage and scalability of LoRa LPWAN for internet of things," in *Proceedings of the 2018 IEEE/ACM 26th International Symposium on Quality of Service, IWQoS 2018*, June 2018.

[27] F. Adelantado, X. Vilajosana, P. Tuset-Peiro, B. Martinez, J. Melia-Segui, and T. Watteyne, "Understanding the limits of LoRaWAN," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 34–40, 2017.

[28] M. Cattani, C. A. Boano, and K. Römer, "An experimental evaluation of the reliability of LoRa long-range low-power wireless communication," *Journal of Sensor and Actuator Networks*, vol. 6, no. 2, p. 7, 2017.

[29] XAMPP, Apache Friends, https://www.apachefriends.org/es/index.html.