

App Based System for Monitoring of Time Schedule in Public Transportation Using Node – MCU

Gowtham. B¹, Dharneesh. R², Jeevananthan. S³,
Dr.Mangalam. H⁴

^{1,2,3}Student , Electronics and Communication Engineering, Sri Ramakrishna Engineering College,
Coimbatore,Tamil Nadu

⁴Professor ,Electronics and Communication Engineering, Sri Ramakrishna Engineering College, Coimbatore,
Tamil Nadu

Date of Submission: 01-01-2023

Date of Acceptance: 10-01-2023

ABSTRACT

In Public transportation is now among the most crucial elements in the development of cities and the enhancement of people's quality of life. The most significant issue for those who use public transit is the lengthy wait at the bus station. The suggested system enables users of public transit to use a mobile application to find public transportation, arrival schedules, and other information from any location at any time. Using RFID and the Node MCU, the system was created using Internet of Things (IoT) technologies. A tracking gadget and a web-based monitoring application are designed as a monitoring system. The graphical user interface (GUI) of a web application is created to aid users by tracking the IN and OUT times of public transportation. The real-time monitoring of measured sensor outputs as well as the relay on/off status shown over the Blynk App combined with the real-time controlling of relays validated the work. The software has been handled for the management, controlling, transaction reporting time of the system and hardware. Here, attempts were made to fully utilise the capabilities of the cloud-based mobile app Blynk, which was created expressly for Internet of Things (IoT) applications. In order to accomplish this, a Node MCU-based Internet of Things system was created. It allows for real-time monitoring of the output measurements from numerous sensors and also gives the user the ability to regulate the electrical load that is linked to it.

Keywords: Public transportation, RFID and the Node MCU, Internet of Things (IOT), Blynk, graphical user interface (GUI), real-time monitoring, mobile application

I. INTRODUCTION

When we talk about public transportation, we're talking about shared passenger vehicles like buses, trolleybuses, trains, ferries, and express vehicles like the metro. Smart Public Transportation is a component of Intelligent Transportation Systems (ITS) (SPT). Public transportation networks can be monitored intelligently to ensure their functionality and to educate customers about outings and system operational conditions. smart citizens with the help of different technologies and the design of an intelligent real-time public transportation monitoring system based on IoT transport systems, SPTS is able to manage and regulate the transportation network as well as retrieve data from various sensor systems.[1] There are numerous cutting-edge technologies that have aided in the creation and adoption of intelligent public transportation systems, including geographic information systems (GIS). A network of linked physical items known as the Internet of Things (IoT) can be accessed online. Previous studies have demonstrated how many wireless technologies can control and monitor smart public transit systems including RFID, node MCU, and Wi-Fi, among others, by merging information technology, cutting-edge methodologies, and smart sensing systems. A network of "connected things" is known as the Internet of Things (IoT). The majority of the time, the items have an embedded operating system and a communication interface for the internet or other nearby things.

An IoT service is one of the essential components of a general IoT system that connects

the numerous 'things'. The 'items' that make up the IoT systems have an interesting implication: they are powerless to act on their own. They ought to be able to connect to other 'things' at the very least. Passive tags can only be used in an area that is close to the RFID reader since they need a strong RF field to function. This is because there is a lack of information regarding the bus's estimated time of arrival (ETA), current location, and occupancy (number of seats available). [2] In order to improve customer happiness and ridership, particularly for university buses, real-time transportation information systems play a critical role. Therefore, it is important that the public transport like buses are monitored and tracked so that users are informed of the bus's current location and the bus's ETA. Real-time bus tracking for improved transport management is now possible because to the development of Radio Frequency Identification and Open-Source microcontroller technology. These technologies can be applied to public transport systems, especially buses, which could be possible to make the monitoring and guidance for the 19 proper time-scheduling for the public bus transportation network.

II. SYSTEM STUDY

EXISTING SYSTEM

In the current system, almost all public transportation systems require monitoring units to check the arrival and departure times of buses. However, someone still needs to keep an eye on how public transportation is scheduled, and human mistake is always a possibility.

DRAWBACKS OF EXISTINGSYSTEM

- To examine and update the database, the huge amount of working hours and human labour are required.

- High privacy concerns, a lack of time uncertainty, and limited flexibility might be required.
- Public transportation entry and exit times can be collected manually with a variety of mistakes and storage options

PROPOSED SYSTEM

In the suggested system public transportation monitoring system, has built and developed by using various hardware and software like(RFID, NODE MCU,BLYNK ect..) which is been connected to an IOT application over an internet service .

ASPECTS OF THE PROPOSED SYSTEM

- With its limited power and conception of time, it minimises human labour and effort.
- Using the data, it receives from the Blynk server and the information the user has provided in the android application,the android programme provides the necessary data to the user.
- Open-source software enables access from anywhere at any time, enabling employees to store data as digital records for feature references.

III. BLOCK DIAGRAM

A power supply RFID reader, an RFID card that supplies power to the Node MCU so it can display the entry and exit times through LEDs, and an IOT application make up the block diagram of the public transportation monitoring system using the Internet of Things (IoT) illustrated in (Fig name 1).

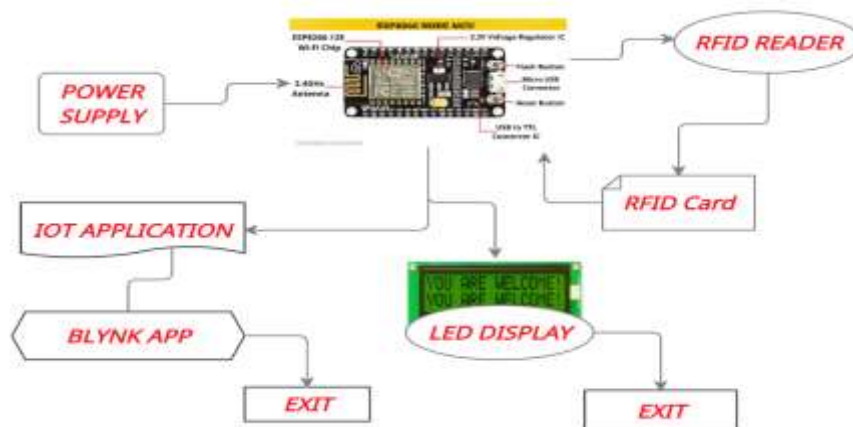


Fig name 1: Block Diagram of Public Transportation Monitoring System using IOT Application

IV. PROBLEM DEFINITION

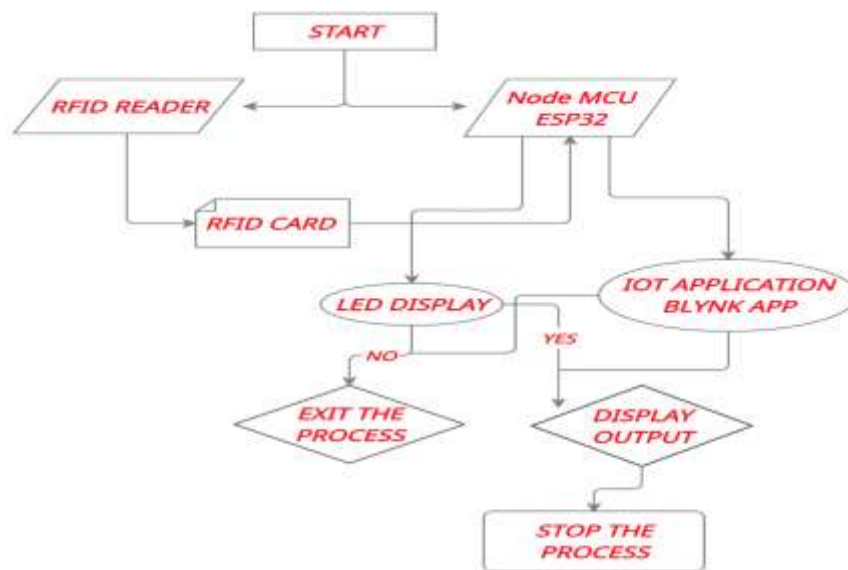
While it is quick and practical mode of transportation, public transportation has a lot of drawbacks. The security evaluations that are being performed at numerous transport systems could normally be used to define the issue. Roadways and transportation systems are becoming increasingly busy as demand for transportation outpaces capacity growth in many locations. It is getting more and more crucial to offer transit-friendly techniques that can shorten travel times, increase dependability, and save money on operating costs. [3] For the purpose of offering transportation solutions that are financially viable, transportation management techniques that maximise the capacity of already-available resources must be investigated. This study offers a comparative examination of all the cutting-edge existing concepts while keeping in

mind the necessity for an intelligent transportation system. Real-time and historical data are typically needed to track the vehicles. MCU, RFID, and Internet of Things (IoT) devices, among others, are utilised for real-time nodes. Data analytics and machine learning are also employed for better vehicle scheduling as a result of the production of enormous amounts of data from IoT enabled devices existing in the transportation system.

V. FLOW CHART

Flow chart of public transportation monitoring system in (fig name 2): Based on the course of events After inserting the RFID card, the RFID reader sends the information to the Node MCU, which analyses it and uses it to display both entry and exit times on the LED display and in the Blynk app.

Fig name 2: Flow Chart of Public Transportation Monitoring System with Node MCU and IOT Application



VI. HARDWARE DESCRIPTION

Node MCU

A LUA-based open-source firmware called Node-MCU was created for the ESP8266 wi-fi chip. Node MCU firmware is included with the ESP8266 Development board/kit, also known as the Node MCU Development board, in order to explore ESP8266 chip capability. Since Node MCU is an open-source platform, anyone can edit, tweak, or manufacture its hardware. The ESP8266 wi-fi-enabled chip is part of the Node MCU Dev Kit/Board. Expressif Systems' ESP8266 is a low-cost Wi-Fi chip that uses the TCP/IP protocol. The ESP8266 Wi-Fi Module can be used to learn more about the ESP8266. There is a Node MCU Development Board v1.0 (Version2), which is

available for the Node MCU Dev Kit and typically has a black coloured PCB.

RFID READER

The two components of the digital network known as Radio Frequency Identification are tags and readers (RFID). The reader is a technical device that transmits radio waves and receives signals from RFID tags using one or more antennas. Tags can be passive or active, transmitting their identity and other material to nearby readers using radio frequencies. Without a battery, passive RFID tags are powered by the reader. Active RFID tags are stored in a battery.

LCD DISPLAY

The term "Liquid Crystal Display," or LCD, applies to a flat display technology that is typically seen in computer displays, instrument panels, smartphones, digital cameras, TVs, laptops, tablets, and calculators. It is a thin monitor that supports high resolutions and provides superior image quality. LCDs have begun to replace the more antiquated CRT display technology, and new display technologies like OLEDs are beginning to do the same.

RFID CARD

Radio-frequency identification is referred to as RFID. Applications requiring access control or the need to track or identify personnel often make use of RFID cards. Cards currently use a variety of RFID frequency bands, including 860-960 MHz ultra-high frequency, 13.56 MHz high frequency smart card, and 125 kHz low frequency proximity (UHF).

CIRCUIT BOARD

The circuit boards in an electronic device are what make it work. They mechanically sustain the product while connecting all of its electrical components. Each electrical component that makes up a printed circuit board (PCB) is created during the manufacturing process and is crucial to the functionality of the device to which it will be linked.

ADAPTOR

Any hardware or circuitry that permits the communication between two different types of devices is referred to as an adapter in general. Hardware adapters could be "onboard" your computer (soldered directly onto the motherboard). They could also be "discrete" (separate hardware, attached to a port on the motherboard, so that they can be removed or replaced).

VII. SOFTWARE DESCRIPTION

Arduino IDE

To write code and upload it to Arduino boards, utilise the open-source Arduino IDE programme. The IDE application is suitable for a variety of operating systems, including Windows, Mac OS X, and Linux. It supports the programming languages C and C++. This phrase alludes to an Integrated Development Environment. In the Arduino IDE, writing a programme or piece of code is frequently referred to as "sketching." We must connect the Genuino and Arduino board with the IDE in order to upload the sketch that was developed in the Arduino IDE programme. The sketch's file extension is ".ino."

Sketches are computer programmes created using the Arduino Software (IDE). These drawings are created in a text editor and saved as files with the ".ino" extension. The editor offers functions for text replacement and text searching. When saving and exporting, the message section provides feedback and shows errors. Our on-screen code is compiled and executed when you click the Upload button. The code is then uploaded to the connected board.[4] We must confirm that the appropriate board and ports are chosen before uploading the sketch. To link the board to the computer, we also need a USB connection. After completing the aforementioned steps, select the Upload button from the toolbar.

The most recent Arduino boards provide an automated reset feature before starting an upload. We must push the Reset button that is available on the earlier boards. The Tx and Rx LEDs will begin to blink as soon as the uploading is complete. The notice will appear in the error window if the uploading fails.

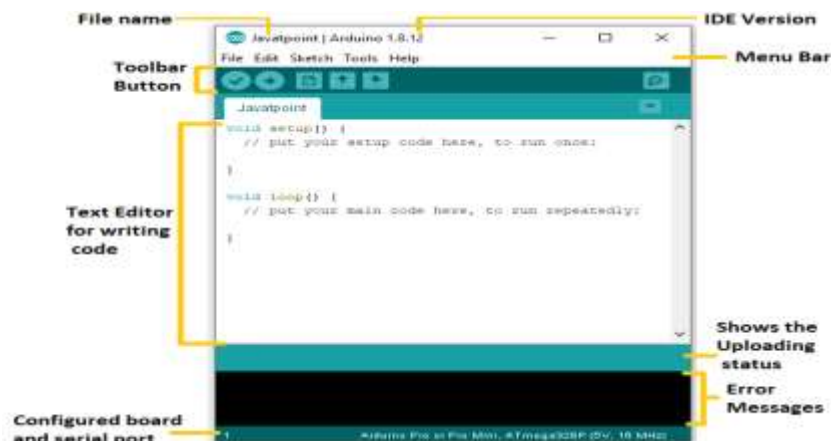


Fig name 3: The Arduino IDE will appear

The console shows text generated by the Arduino Software (IDE), including error messages in their entirety and other data. The configured board and serial port are visible in the window's bottom right corner. You may create, open, and save sketches, validate and upload programmes, open the serial monitor, and more using the toolbar buttons.

Arduino IDE - Installation

Step 1 : Requires a USB wire and an Arduino board (you can choose the board of your choice).

If you use an Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B connector), the kind you would use to connect to a USB printer.

step 2: The Arduino IDE software must be downloaded.

You can download several versions of the Arduino IDE from the Download page of the official Arduino website. Software that is compatible with your operating system must be chosen (Windows, IOS, or Linux). When the file has completed downloading, unzip it.

Step 3: Turn on your board's power.

The USB port on the computer or an external power source are used automatically by the Arduino Uno, Mega, Duemilanove, and Nano. Make sure the Arduino Diecimila is set up to receive power from the USB connection if you're using one. A jumper, a little piece of plastic that attaches to two of the three pins between the USB and power connections, is used to choose the power source. Make sure it is connected to the two pins that are closest to the USB port.

Utilize the USB cable to connect the Arduino board to your PC. The green power LED, which is marked PWR, ought to shine.

Step 4 :Involves starting the Arduino IDE.

After downloading the Arduino IDE application, you must unzip the folder. Inside the folder, you may find the application icon marked with an infinity label (application.exe). Double-click the icon to start the IDE.

Step 5: start your first project.

Soon after the software launches, you have two options: Make a fresh undertaking. Visit a project that is currently under progress.

step 6: select your Arduino board.

To avoid any issues when uploading your application to the board, you must select the proper Arduino board name that corresponds with the board linked to your computer. Go to Tools Board to choose your board.

step 7: select your serial port.

Select the serial device for the Arduino board. Choose Tools > Serial Port from the menu. This most likely has a COM3 or above (COM1 and COM2 are usually reserved for hardware serial ports). You can verify by unplugging your Arduino board and reopening the menu; the entry for your Arduino board that disappears should be the one you need. When you reconnect the board, select that serial port.

Step 8: Is where you upload the programme to your board.

Before discussing how to upload our software to the board, we must first demonstrate how each symbol in the Arduino IDE toolbar functions.

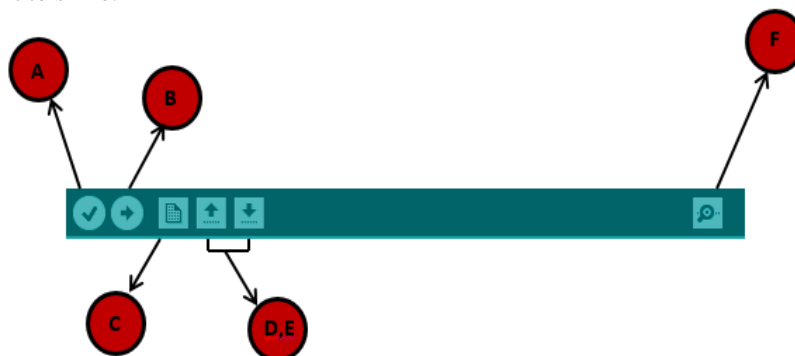
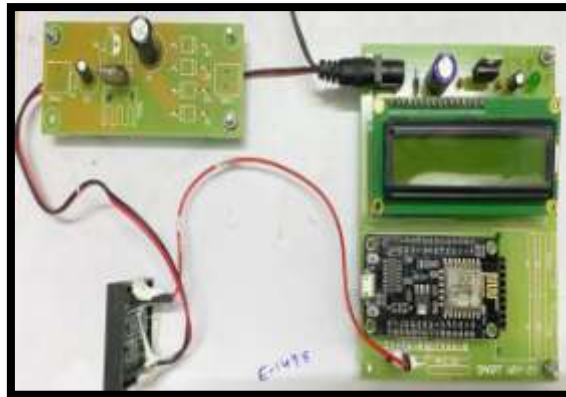


Fig name 4: symbol appearing in the Arduino IDE toolbar

- **A**-check to see if there are any compilation errors.
- **B** -applied during the programming of an Arduino board.
- **C** -is the shortcut for beginning a new drawing.
- One of the example sketches can be opened directly by pressing **D**.
- Save your drawing using **E**.
- **F** - Serial data is sent to and received from a circuit board via a serial monitor.
Just choose "Upload" from the environment's menu at this time. After a little delay, the board's RX and TX LEDs will begin to flash. If the upload is successful, "Done uploading" will appear in the status bar.

VIII. OUTPUT

Step 1 : The connections between each of the hardware components are shown in the below images.

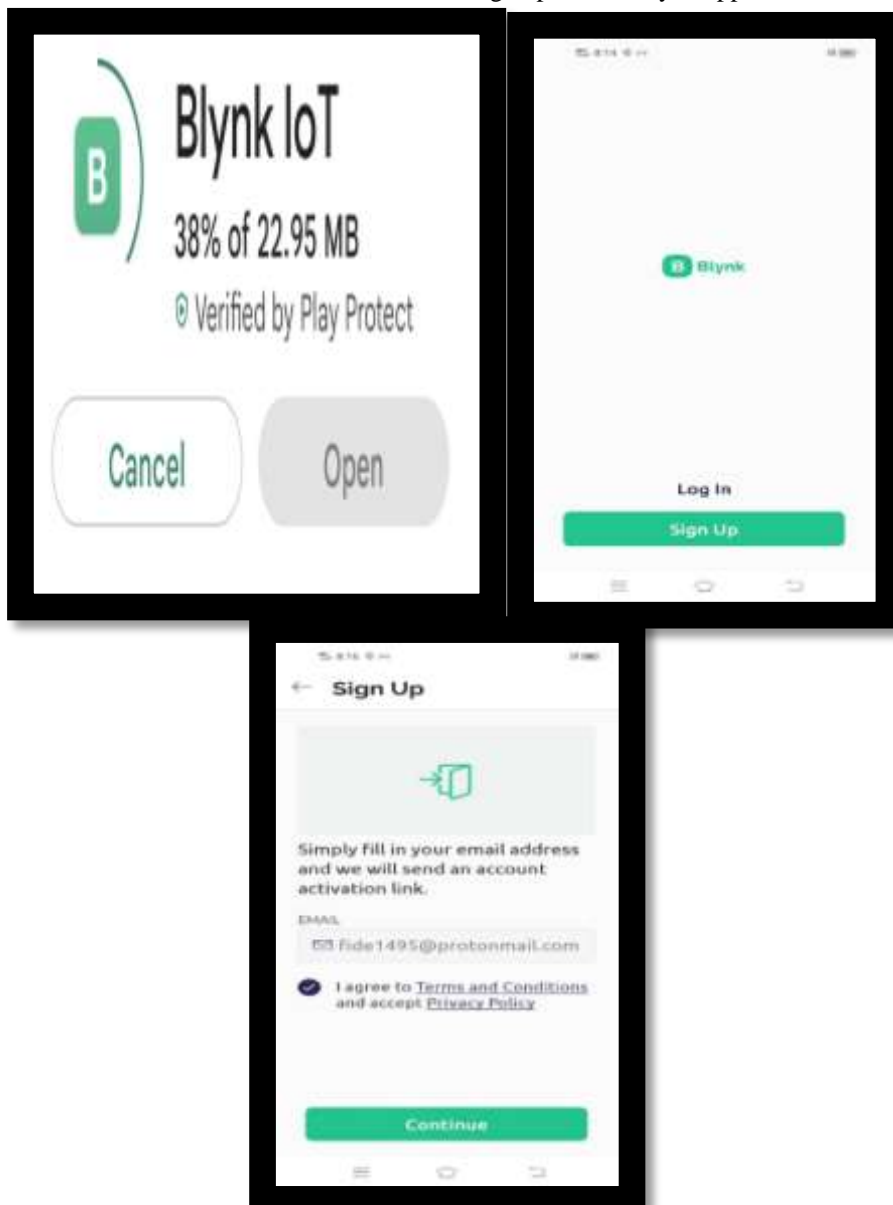


Step 2 :A public transportation entry and exit time appears in the LCD screens below.





Step 3: The screenshots below illustrate how to instal and sign up for the Blynk application.



Step 4: The images below display the arrival and leave times for public transportation using an IOT application (BLYNK app).



IX. CONCLUSION

This paper has given readers a look into the current status of ITS technology, with a focus on applications linked to public transportation. Although it has made an effort to show how these applications may be used to enhance public transportation, both on a service-by-service and network-wide level, as well as the effectiveness of the transportation system overall for predicting entry and exit times for public transportation, there are still a number of problems to be resolved. Predicting public transportation time schedules is still an unresolved problem. To address these challenges, many researchers have employed various methodologies. We have discussed these problems, the techniques researchers have taken to address them, and the drawbacks of their efforts. We have also spoken about what we can do to enhance the public transportation system's timing.

X. FUTURE ENHANCEMENT

However, as this research has shown, it's possible that in the future, rather than isolated uses of certain ITS technologies, the biggest benefits may come from an integrated development of ITS with the ultimate goal of building a fully intelligent transport network. Future research should focus on hybrid communication between devices and mechanisms that increase the amount of secrecy in the design of public transportation systems. We can also employ additional hardware and software tools to enhance the transportation system.

BIBLIOGRAPHY

- [1]. BLYTHE P.T. ^[et] (1994) Smart Card Applications in Transport. International Smart Card Industry Guide, Published by Smart Card News, UK.
- [2]. Sulaima Lebbe Abdul Haleem ^[et] , Samsudeen Sabraz Nawaz ^[et] , "Real Time

- Bus Tracking and Scheduling System Using Wireless Sensor and Mobile Technology", Journal of Information Systems & Information Technology (JISIT) Vol. 1 No. 1 2016 , ISSN: 2478-0677s
- [3]. HICKMAN, M D^[et] and WILSON, N H^[et] (1995) Passenger travel time and path choice implications of real-time transit information Transportation Research "C" Vol. 3 Number 4 pp 211-226
- [4]. Dinkar R^[et] ,Patnaik Patnaikuni^[et], A Comparative Study of Arduino, Raspberry Pi and ESP8266 as iot Development Board, International Journal of Advanced Research in Computer Science, Volume 8, No. 5, pp. 2350-2352, May-June 2017
- [5]. Komal Satish Agarwal^[et], Kranti Dive^[et], "RFID Based Intelligent Bus Management and Monitoring System", International Journal of Engineering Research & Technology (IJERT), IJERT, ISSN: 2278-0181, www.ijert.org, Vol. 3 Issue 7, July – 2014
- [6]. Sutar, S.H. ^[et], Koul, R. ^[et] and Suryavanshi, R. ^[et] (2016) Integration of Smart Phone and IOT for Development of Smart Public Transportation System. International Conference on Internet of Things and Applications, Pune, 22-24 January 2016, 73-78. <https://doi.org/10.1109/IOTA.2016.7562698>
- [7]. Karthikeyan G [et], Jawahar M L[et], "Smart Bus Management System", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, Special Issue – 2018
- [8]. Jananey, J., ^[et al.] (2019) Review on iot Based Architecture for Smart Public Transport System. International Journal of Applied Engineering Research, 14, 466-471. <http://www.ripublication.com>
- [9]. Nihad Hasan Talib^[et], "Real - Time Transportation Monitoring Using An Integrated RfidGis Scheme ", International Journal of Mechanical Engineering and Technology (IJMET), Volume 9, Issue 13, December 2018, Article ID: IJMET_09_13_001, ISSN Online: 0976-6359
- [10]. Apsara, S. ^[et], Rashmi, G.A. ^[et], Mohan Kumar, K.V. ^[et], Anitha, L. ^[et] and Jyothi, B. ^[et] (2017) Internet-of-Things (IOT), Navigation System (NS), Micro Navigation (MN), Crowd-Aware Route Recommendation (CARR), Bus Arrival Time Estimation (Bate). International Journal of Advance Research, Ideas and Innovations in Technology, 3, 1163-1170. <https://www.IJARIT.com>