Parking Management System Using Internet of Things

Aditya Sarin and Deveshi Thanawala

Abstract The raid population growth and increasing number of private vehicle ownership in countries like India have largely contributed to vehicle congestion. Inappropriate vehicle parking areas are considered as one of the root causes for slow moving traffic and has invoked the attention of the administration officials. The model proposed here serves a three-stage parking system using the Internet of Things. The first stage consists of an automated gate connected to an onscreen vehicle counter. The second stage consists of sensors that are installed at each parking slot, connected with a cloud platform by using a microprocessor board. These sensors send real-time data to the cloud by keeping track of the status of each parking slot and provide an online platform for the user to check the availability of parking slots. The final stage of the system provides an application interface that sends the user's parking details through e-mail and displays the same on the application.

Keywords Parking management system · Internet of Things · Cloud management · Parking security management

1 Introduction

Private vehicle ownership has engendered a major setback to the public transport community in India resulting in a vast proliferation of private automobiles over the last two decades. The number of registered vehicles in India has increased from 55 million in 2001 to 210 million in 2015 and is subject to escalate [1]. Being the second most populated country in the world these new private automobiles have also incurred a setback in its development.

A. Sarin (✉) · D. Thanawala
Mukesh Patel School of Technology Management and Engineering, Electronics and Telecommunication, Mumbai 400054, India
e-mail: aditya28sarin@gmail.com

D. Thanawala
e-mail: deveshi.m.thanawala@gmail.com

V. Suma et al. (eds.), *Evolutionary Computing and Mobile Sustainable Networks*, Lecture Notes on Data Engineering and Communications Technologies 53,
https://doi.org/10.1007/978-981-15-5258-8_34

One of the primary effects due to this population explosion is space management that has given rise to an inevitable problem of vehicle parking. Vehicle management at the parking lots has become an important aspect of the best utilization of existing parking area capacity. Wastage of time and fuel at the parking area due to an increase in vehicle count has become a major issue for everyone [2].

Internet of Things (IoT) relates to a vast community of devices connected to the Internet which form a broad network where data from these billions of devices are shared [3]. The main purpose of developing an IoT-enabled parking management system is to put an end to the endless parking area problems such as finding availability of slots and the irregularity of entry and exit of the vehicles that cause congestion in such parking lots.

The proposed system works in three different stages. The first stage is designed to provide an automatic gate control mechanism that detects the presence of any vehicle and lets it in via the automatic gate. Once the vehicle has entered, an LCD screen keeps track of the number of vehicles that has entered. The second stage comprises the parking slots being enabled with sensors that detect if a vehicle is parked or not, sending this data to the cloud server that can then be monitored by the security officials in the control room. The security personnel can then track the unwanted congestion of vehicle movement inside the parking space by keeping a check on the number of vehicles entering and exiting the parking lot, which has to be same at all instants with a delay of some minutes acceptable for a car to move out of the parking lot. The third stage displays the cloud data on a mobile application and also sends an e-mail to the concerned vehicle owner about his parking details.

The structure of the paper is as follows: Sect. 2 explains the literature survey. Section 3 describes the system layout along with hardware description and the design flow which derives a stepwise procedural order in which the model works. Section 4 enumerates results and discussion. Section 5 includes the conclusion.

2 Related Work

Outdoor parking slots are available with GPS to keep a track on the position of the vehicle whereas for indoor parking slots Wi-Fi-enabled signatures were used to provide a similar mechanism [4].

Several models have been proposed to enable vehicle detection by interfacing sensors at the parking site [5]. Alternate solutions have been provided using RFID technology to automate car parking systems [6]. A large number of parking models have been proposed and implemented using Bluetooth, wireless networks, Zigbee, and RFID technologies. Among such works, the idea proposed in paper [7] explains the use of RFID for a smart parking system where the parking slot availability is managed by using RFID and periodically saved in the database.

Paper [8] provides an efficient interface for user interaction but uses infrared sensors for detection at a parking spot but such sensors have a low range of detection (about 20 cm). However, the paper gives an additional solution using advanced driver assistance which provides an efficient algorithm for charging the driver on the basis of the time for which the car is parked.

Paper [9] provides a solution for parking challenges being faced in Hyderabad by providing online booking of parking slots along with entry based on a barcode scanner. Paper [10] explains the usage of smart parking with the help of Zigbee communication and interfacing IR sensors for the detection of vehicles also using RFID technology. Paper [11] uses MQTT protocol which provides three levels of security levels on top of the TCP layer. It also integrated temperature sensors to enable the safety of the parking lot. Ultrasonic sensors are used for indoor parking and IR sensors are used to keep a check in outdoor area parking, sending this data to a web page.

Paper [12] provides a comparative study of different methods that are employed in improving exiting parking conditions, it provides methods such as using ultrasonic sensors, managing parking status using the online cloud, and RFID to detect the vehicles and sensor nodes. Paper [13] developed a queuing mechanism additionally which works in a first-in first-out manner, which is structured for peak hour and non-peak hours timing as well, providing statistical parameters to function in a multiple queue system. In Paper [14], the authors explain a model involving an Android application to ease the parking slot booking system and used a deep learning algorithm to detect the vehicle number plates.

3 Proposed Work

The proposed model illustrates a modular design of the parking management system integrating the hardware components with the software architecture. Figure 1 describes the system layout proposed that forms the foundation of the model illustrating the location and working of each device.

3.1 System Layout

3.1.1 Microprocessor

The microprocessor forms the core of the model that is primarily concerned with the sensor data being uploaded to the online cloud platform. It updates the sensor data and keeps the users updated from time to time. The microprocessor used is a Raspberry Pi B + model. Three ultrasonic sensors are connected that detect the presence of the vehicle in each parking spot.

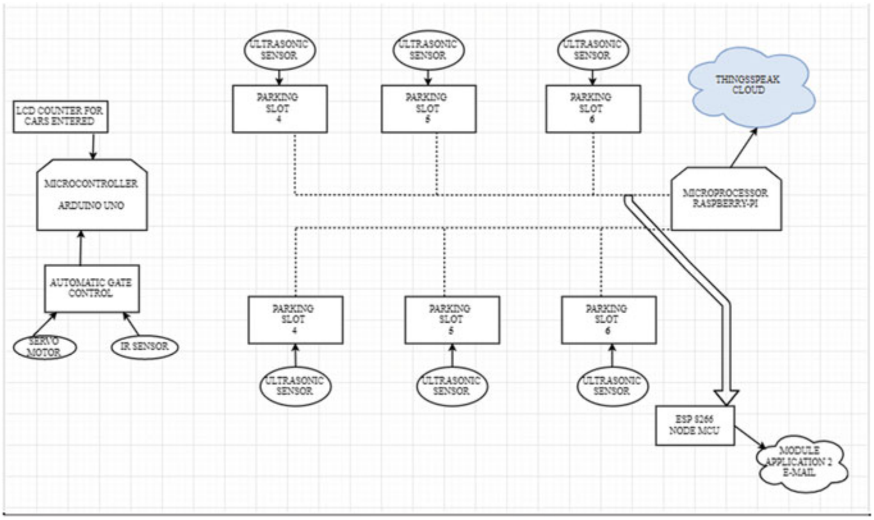

Fig. 1 System layout

3.1.2 Embedded Microcontroller

The microcontroller forms the nucleus of stage 1 of the proposed model that controls the working of the automatic gate control as well as the counter display at the entry of our parking lot. This board is connected to the IR sensor, servomotor, and LCD display. It facilitates the entry of the vehicle inside the parking lot and keeps track of the number of vehicles entering. The microcontroller board used in the proposed model is an Arduino UNO based on the microchip ATmega328P.

3.1.3 Node MCU ESP8266

Node MCU works on the ESP8266 Wi-Fi system-on-chip firmware. This module is connected to an LED at each parking slot indicating that the slot is filled. This data is then displayed on the BLYNK application.

3.1.4 Counter

The counter uses an LCD display which is present at the entry gate of the parking lot displaying the number of vehicles entering the area thereby notifying the driver if any parking lot is available or not thus preventing unnecessary traffic inside the parking lot.

3.1.5 IR Sensor

This sensor transmits an infrared wave from its transmitter terminal which hits the obstacle and returns at its receiver terminal, and hence it can determine the presence of an entity. IR sensor emits infrared radiation and helps in detecting the presence of an obstacle. However, an ultrasonic sensor emits ultrasonic sound waves that can accurately determine how far the obstacle is.

This sensor is not used as a parking sensor because IR sensors are highly sensitive to dust, smoke, and light. Moreover, IR sensors have a small detection range, and hence it can be used at the entry where vehicles move close to them but could fail to detect vehicles with their base at a greater height from the ground.

3.1.6 Servomotor

The servomotor forms the mechanism of movement for the entry gate, this servomotor is connected to the microcontroller and is programmed to provide rotation by 90° in clockwise as well as in counterclockwise direction.

3.1.7 Ultrasonic Sensor

This sensor detects the vehicle when the ultrasonic waves are blocked also calculating the distance at which the obstacle is present. It is placed with the sensor's transmitter and receiver facing upward, installed at the center of each parking spot. The orientation of the sensor is illustrated in Fig. 2. The sensor has a maximum range of 4 m and a minimum range of 2 cm with an angle that can be varied by 15°.

$$D = (V \times t)/2 \tag{1}$$

Formula 1 is used to calculate the distance between the sensor and the obstacle.

where

- D required distance,
- V speed of sound, and
- t time sound wave takes after striking the object.

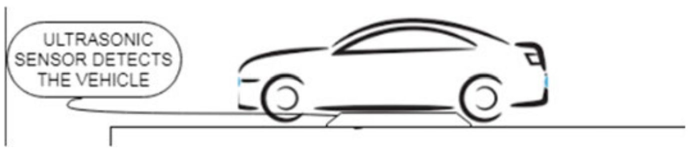

Fig. 2 Ultrasonic sensor orientation

The distance "*D*" is calculated using the above formula and this is how the sensor calculates the distance between an obstacle and itself.

3.1.8 ThingSpeak

ThingSpeak is an open-source cloud platform designed for the Internet of Things application developed by Mathworks. This online platform lets us store and collect such sensor data and interact with different forms of data stored. We can either keep our data public or private.

3.1.9 Blynk Application

Blynk is a user application specially designed for the Internet of Things. It is used to control hardware remotely, display sensor data, and also to store the data. In the proposed model, Blynk is used as an application through which the users can keep a track on parking availability. The application also provides notification service to the user via e-mail stating the location of the user's parked vehicle.

3.2 Design Flow

Figure 2 above explains the flow of system architecture.

The entire flow can be described in four steps:

Step 1

The vehicle arrives at the parking lot entry, the driver can then check for the availability of parking spaces on the LCD counter, if any. If there are empty parking slots the vehicle enters the area through an automated gate that is a combination of an IR sensor and a servomotor connected to the Arduino Uno microcontroller. This microcontroller programs the threshold of the IR sensor. The sensor detects the entry of the vehicle that then sends signals to the microcontroller. After receiving the signal, the servo gate rotates and opens the entry for the incoming vehicle, and simultaneously the LCD counter updates the number of vehicles entered in the parking space (Fig. 3).

This forms Stage 1 of the proposed model.

Step 2

Here the vehicle that has already entered moves to the vacant parking spot. Each parking slot has an ultrasonic sensor engraved that detects if a car is parked or not. The sensor is programmed to a preset threshold. The transmitter sends the ultrasonic wave which is then received at the receiver after being reflected from the underside of the vehicle. If the distance of the obstacle is less than the given threshold, the sensor gets triggered and sends data accordingly.

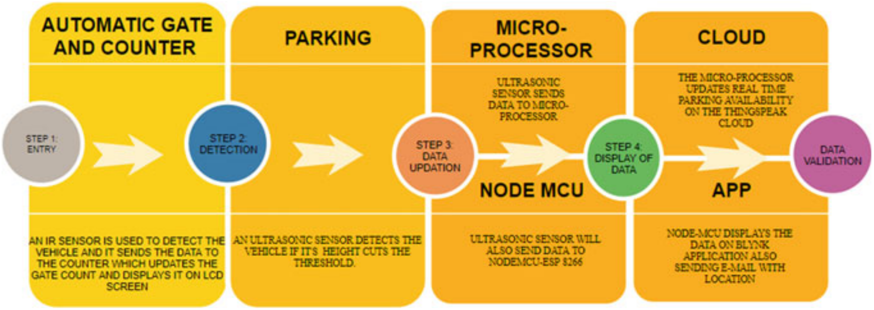

Fig. 3 Design flow

Step 3 and Step 4

Step 3 and 4 are further divided into two parts.

Path 1

The data from the sensor is sent to the microprocessor which runs an algorithm to calculate the distance between the obstacle and the sensor with the help of the time taken for the wave to return. This data is then sent to an online cloud platform where it displays the number of parking slots and even the real-time parking availability so users can keep a check on the availability of parking slots conveniently. If the distance falls under the pre-determined threshold then the indicator on the cloud indicates that the particular slot is filled. The cloud platform used is the ThingSpeak cloud platform.

Path 2

The data from the sensor activates the on-site led which sends data to the Node-MCU ESP 8266 module. This module is programmed to connect with the Blynk application where the parking chart is displayed in a similar manner to the cloud platform, which provides users an alternate way to keep a check on the availability of the parking slots. The application is also programmed to provide an additional feature in which the user receives an e-mail notification as soon as he parks his car, specifying the users' vehicle location (block number and the floor number).

3.3 Logic Flow

Figure 4 explains the algorithm for vehicle detection using an ultrasonic sensor and sending data to the server.

The proposed model describes an algorithm for the operation of ultrasonic sensor. The sensor is first initialized by keeping the trigger pin low, this pin is an output pin

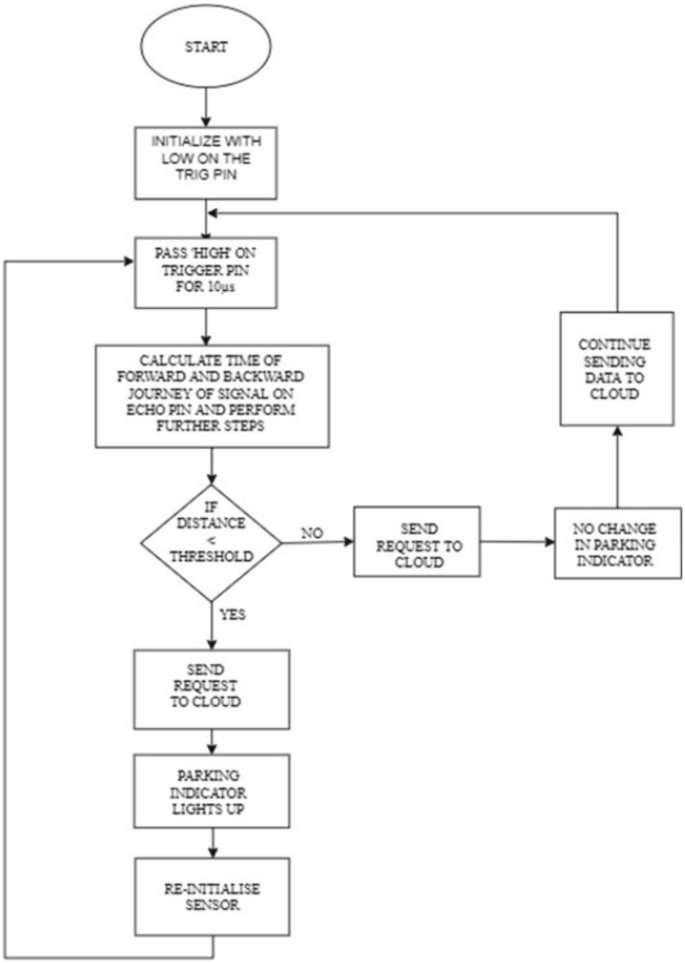

Fig. 4 Logic flow of ultrasonic sensor detection

and sends ultrasonic signals. The trigger pin is then made high for 10 µs for the signal to transmit. After 10 µs the trigger pin is made low again.

Now a threshold is set in the sensor, suppose "*x*", this threshold is set at 25 cm. If a car with a ground clearance of 10 cm is parked, the sensor detects an obstacle within its threshold range and indicates a deviation from its ideal state. The cloud platform detects a change in the sensor state and indicates the same on the screen. This demonstrates that the parking slot is occupied. The indicator turns off when the car leaves the parking slot, and hence the sensor has no obstacle in its threshold range.

Figure 5 explains the algorithm for vehicle entry using gate control and counter.

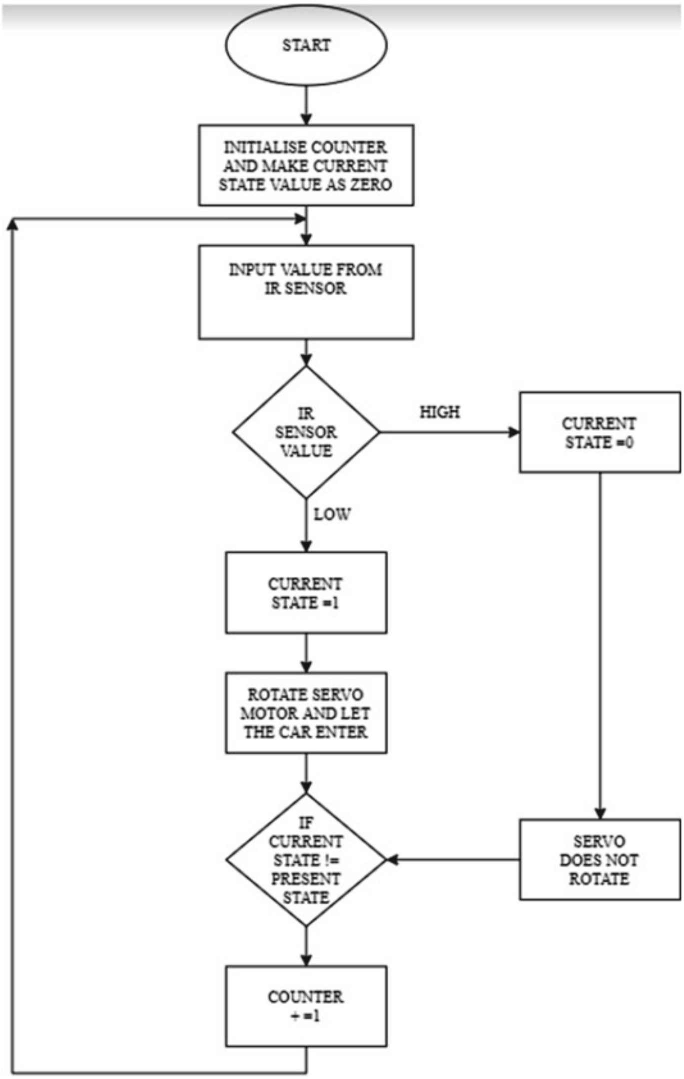

Fig. 5 The logic flow of stage 1

Stage 1 of the proposed model consists of the counter, automatic gate control, and an IR sensor.

The IR sensor emits infrared radiation continuously. A counter variable is created that keeps track of the number of vehicles entering the parking lot. The counter has current as well as a previous state variable. These variables are initialized first and then the input from the IR sensor is taken. A preset threshold is applied to the IR. As soon as the microcontroller board receives a signal when the obstacle is detected, the current state increments by one and the microcontroller sends a signal to the servomotor specifying the degree of rotation it has to make.

Figure 6 explains the algorithm for vehicle detection and sends data to Blynk application.

As soon as the ultrasonic sensor detects a change it transmits a signal to the node MCU which acknowledges this change and displays the same on the application and an e-mail notification is sent to the user specifying the parking slot number. There is an on-site led which indicates if a slot is empty or not.

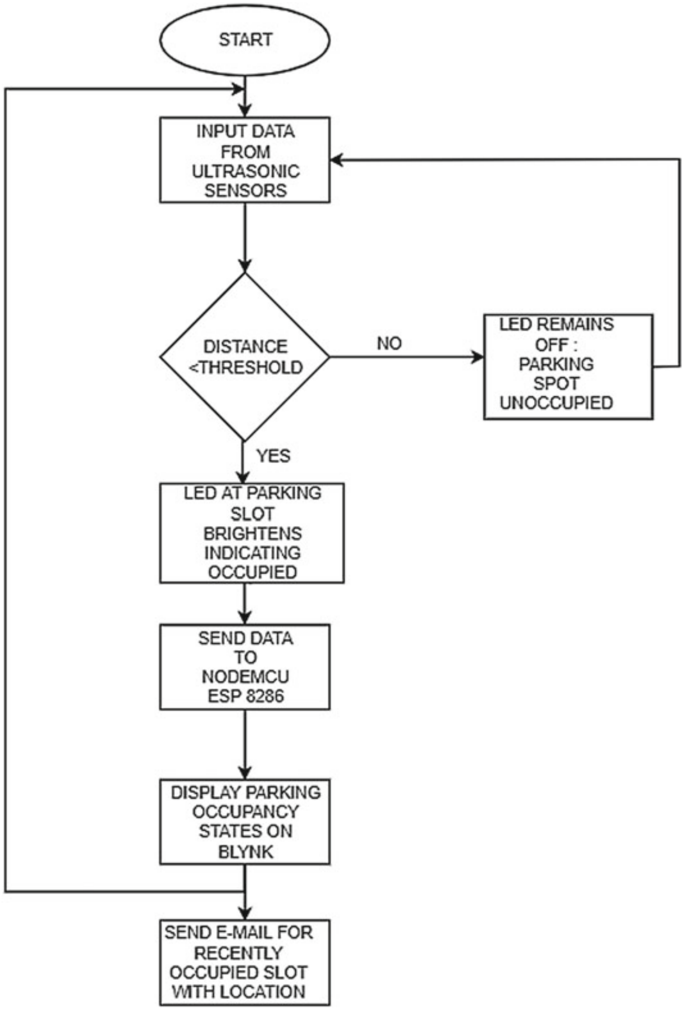

Fig. 6 Logic flow for node MCU with app

4 Results Analysis

The proposed model is designed in such a way that minimum delay is maintained throughout the process to provide users an easier and faster solution to their daily parking problems.

The stage I of the model shows communication between the IR sensor, microcontroller, LCD display, and the automatic gate which occurs in approximately 1.2 s. The prototype for the same is shown in Fig. 7.

The model serves the following advantages:

- It provides early access to the parking lot for users to check parking availability through the cloud platform.
- This model minimizes congestion inside the parking lot as the driver can check for a number of empty parking slots, from the entry gate screen counter.
- Users need not remember the location of their vehicle as an e-mail notification is provided as soon as the user parks his vehicle.
- Security personnel can keep track of the parking history of any particular slot.
- The model also provides a system to keep track of the number of vehicles entering and the number of vehicles being parked at the same time. If the number of vehicles parked is not equal to the number of vehicles that have entered then the security member can infer that a vehicle has left its slot, therefore keeping a better track on vehicle congestion.

Figure 8 shows the parking data displayed on the online cloud platform (ThingSpeak). The data is available in a graphical form as well as in the form of an indicator. The graphical form also displays data history. The indicator signifies the availability of the parking slot. The data takes 15 s to be displayed on the cloud platform.

Figure 9 shows the application interface that displays similar data as on the cloud platform. This data, however, follows its path through the node MCU. The data here signifies two parking slots out of which three are occupied currently.

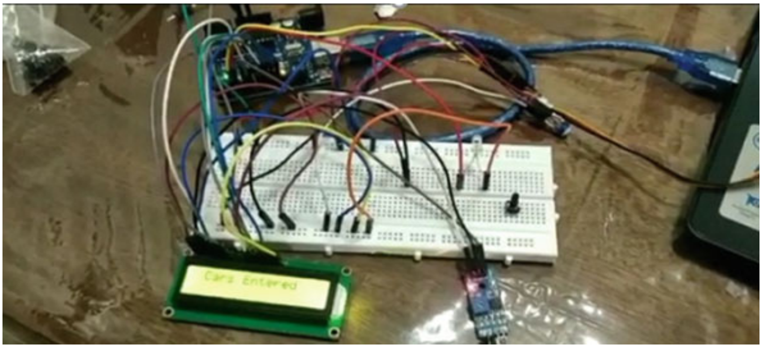

Fig. 7 Prototype for stage I

Fig. 8 ThingSpeak results

Figure 10 shows an e-mail notification that the user receives as soon as he parks his car, in this case, at the P3 parking slot.

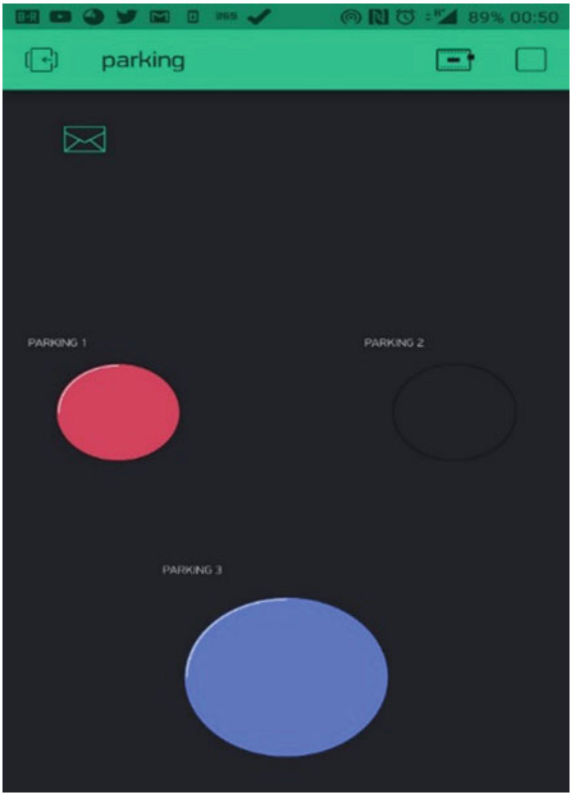

Fig. 9 Output on Blynk application

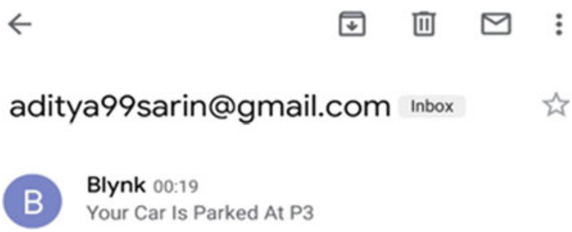

Fig. 10 User receiving e-mail

5 Conclusion

Parking has always posed an additional threat to the interminable congestion on roads and public areas leading to frustration of the driver. This model provides a solution to such a problem with the Internet of Things-based technology to build a parking management system that integrates sensors at the parking site and sends data to the online cloud platform as well as the user application. It provides an e-mail service with vehicle location thus limiting congestion by keeping track of vehicle entry and exit.

The proposed model shows that the time user takes to find his/her parking has been reduced significantly by deploying three sources by which the user can keep track of the parking availability, that is, from an online cloud platform, user application as well as the LCD indicator on-site. The proposed system not only satisfies the user demands but also forms a well-regulated chain of structured data where the security personnel can keep track of their incoming and outgoing vehicles leading to minimum inconvenience.

In the future, the same model can be made more efficient by including an application that can be used for the reservation of parking slots. Along with this, number plate recognition could be used to keep track of the vehicles that have entered the parking space.

References

1. Singh J (2016) Head of the UITP India Office, Public Transport Developments in Indian Cities, Intelligent Transport. https://www.intelligenttransport.com/transport-articles/21458/city-public-transportation-india/
2. Rane S, Dubey A, Parida T, Design of IoT based intelligent parking system using image processing algorithms. In: Proceedings of the IEEE 2017 international conference on computing methodologies and communication (ICCMC)
3. Clark J (2016) What is the internet of things?. https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/
4. Correa J, Katz E, Collins P, Griss M (2008) Room-level wi-fi location tracking
5. Fatima N, Jagtap P, Natkar A, Choudhari ST, Nazish F et al (2018) IOT Based smart car parking system for smart cities. Int J Adv Res Ideas Innov Technol 4(1)
6. Anusooya G, Christy Jackson J, Sathyarajasekaran K, Kannan K (2017) RFID based smart car parking system. Int J Appl Eng Res 12(17):6559–6563. ISSN 0973-4562, ©Research India Publications
7. Hanche SC, Munot P, Bagal P, Sonawane K, Pise P (2013) Automated vehicle parking system using RFID, ISSN (PRINT) 1(2):2320–8945
8. Mahendra BM, Dr. Sonoli S, Bhat N (2017) IoT based sensor enabled smart car parking for advanced driver assistance system. In: 2017 2nd IEEE international conference on recent trends in electronics information & communication technology (RTEICT), India
9. Vakula D, Kolli YK, Low cost smart parking system for smart cities. In: Proceedings of the international conference on intelligent sustainable systems (ICISS 2017) IEEE Xplore Compliant—Part Number: CFP17M19-ART, ISBN:978-1-5386-1959-9
10. Dhumal YR, Waghmare HA, Tole AS, Shilimkar SR (2016) Android based smart car parking system. IJREEIE 5(3):1371–1374
11. Chippalkatti P, Kadam G, Ichake V (2018) I-SPARK: IoT based smart parking system. In: 2018 international conference on advances in communication and computing technology (ICACCT) Amrutvahini College of Engineering, Sangamner, Ahmednagar, India
12. Shanmugasundaram G, Vigneswari T, Abinaya R, Jane Nemisha V, Monisha M, A review on dimension based parking system. In: Proceeding of international conference on systems computation automation and networking 2019, @IEEE 978-1-7281-1524-5
13. Pandey D, Hanchate S (2018) Navigation based-intelligent parking management system using queuing theory and IOT. IEEE. 978-1-5386-5657-0/18/$31.00c
14. Hasan MO, Islam MM, Alsaawy Y, Smart parking model based on internet of things (IoT) and TensorFlow. In: IEEE 2019 7th international conference on smart computing & communications (ICSCC). 978-1-7281-1557-3/19/$31.00©2019