

Parking Management Based on Navigation using Mobile Application management System

Indresh C, Akshitha, Ashwini R Deshi, Vidyashree K, Mrs. Sahana G Kundar

Electronics and Communication Engineering Department, Srinivas Institute of Technology, Valachil
Electronics and Communication Engineering Department, Srinivas Institute of Technology, Valachil
Electronics and Communication Engineering Department, Srinivas Institute of Technology, Valachil
Electronics and Communication Engineering Department, Srinivas Institute of Technology, Valachil
Assistant Professor, Electronics and Communication Engineering Department, Srinivas Institute of Technology, Valachil
Corresponding author: Indresh C

Date of Submission: 26-07-2020

Date of Acceptance: 05-08-2020

ABSTRACT: The expansion and high progress of the Internet of Things have upgraded quality of life and strengthened different areas in society. One of the most popular use cases in smart cities is the implementation of smart parking solutions, because Parking may be considered a problem, especially in the big cities. Careless parking systems are time wasting and cause traffic jams. When a client visits a mall or a centre it may take him/her a long time to locate free parking area. Also more time is needed to pay for parking fees because of the long waiting queues. Accordingly, there is a need to create and design a parking management system using mobile application. This application solves the problems by helping the user to locate free parking slots, pre-booking of slots and money deduction on application depending on the time required. In this system sensors are used to open the gate and to update user about filled and empty slots. Based on the user location nearest person to the parking slot gets the priority. When the user exceeds the allotted time limit additional charges will be deducted from the user account.

KEYWORDS: Arduino UNO, Node MCU, IR Sensor, Servomotor, IR Sensor, cloud database.

I. INTRODUCTION

In the recent epoch, we are in front of a new tricky of parking of vehicles in municipal cities. By using our automated parking system, an operator can save much time for probing free parking space. The user can book in advance the parking slot and keep posted the statistics to the server. Every user has a unique id and keyword. Too many vehicles, too much traffic and there is no enough parking area. This is the situation which is seen in most of the metropolitan

cities today. People keep on roaming on roads searching for a parking space to park their vehicles especially at peak hours of time.



Figure 1. Parking management based on navigation using mobile application

Our projected system presents a smart parking system that controls a number of vehicles to the nearest parking space at any given time based on the parking space accessibility. Our “Smart Parking System” is executed using the Operating System Android. The manipulator requests the Parking Control Unit to check the grade of obtainable parking slots. As soon as the user request, all the accessible free slots are showed to the user in his android phone. If the availability of parking space is confirmed, the user can book the parking slot and continue to pay in advance. One more option is given to the user such that the user can pay the money after using the slot and charges will be based on the time used by the user. The vehicle tracks its path in the direction of the opening of the parking area. The user shots his slots by entering the One Time Password (OTP) which is

sent to him at the parking area. After collaborating the vehicle will further track its path to the assigned parking slot. After successful parking the slot details are updated instantaneously in the administrator's database. Finally, the time to find for an empty parking slot is lessened. The main concern of the "Smart Parking System" is to help the user to find an area where parking is accessible and total number of slots free in that area thus our planned procedure decreases the user's effort and time of searching a parking slot. The parking slot information is uploaded into the centralized database which can be supervised through the smart phones. The users can be registered for their slot in advance through Android Application. Primarily the slot choice is made by the user from his mobile phone. He checks for the availability of a Parking Slot that is adjacent to his location. If it is available, he moves to the next stage or else go to the initial state. Transfers request for parking slot from the mobile using Android application requested by the user. After reserving a particular slot by the user then the status of that respective slot will be marked as RED=RESERVED and the remaining will be GREEN=EMPTY. As soon as the vehicle gets entered into the parking slot, the timer gets ON and measures the total time. As soon as the vehicle moves out of the parking slot, the timer gets OFF and the total cost will be displayed.

1. Proposed Solution

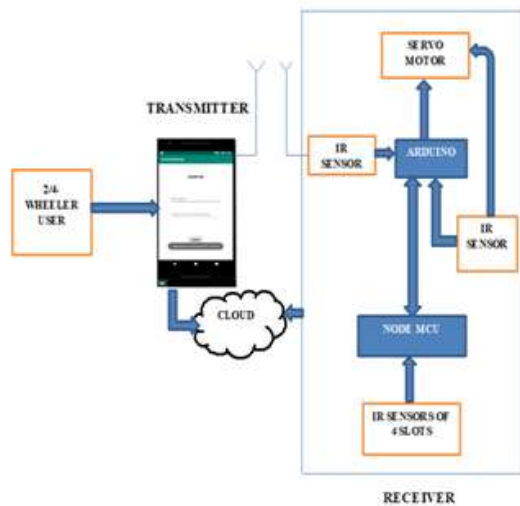


Figure 2. Overview of the project

Transmitter Part:

The simple overview of the block diagram is shown in the figure 2. It comprises of transmitter and receiver parts. The transmitter part contains Android Application and all the statistics are directed to cloud. It can be accessed whenever it's necessary.

Receiver Part:

The receiver part is the area where exactly the parking of vehicle is done. It consists of hardware components. The IR sensor which is kept at the entrance of the parking area and also at the exit side of two gates IR sensors are placed to open the gate. Servo motor is used as gate. The IR sensors are kept near the slots to check the presence of vehicle. All these components are connected to Arduino Uno. Node MCU is also used since it has built in Wi-Fi module and can be interfaced with cloud.

2. Execution Process

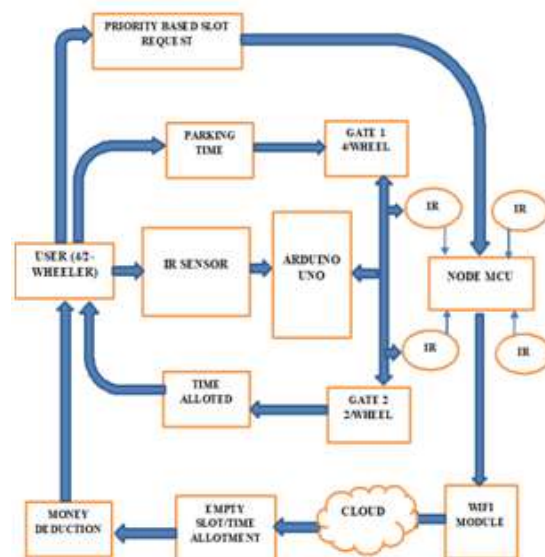


Figure 3. Block diagram

The main block diagram of the project is shown in figure 3. At first the user should register using Android application. In the transmitter part with the help of an Android application the user is requested to book an empty slot in the parking area. The request is priority-based request which means the user near to the parking area will get the first priority to occupy the slot. This is applicable only when one empty slot is left to occupy and also when two users requests for the same slot. The user can also set the required parking time. All the information given by the user are sent to cloud for future reference.

After reaching the parking slot the IR will detect the vehicle, based on the IR sensor the two-wheeler parking area gate or the four-wheeler parking area gate will open. The operation of gate will be completed by servo motor. Before that the user need to enter the code, which will be sent to him after completing the booking process. This is done for the security purpose and for assurance so that no unknown user is allowed. All these operations are done with the help of Arduino which is a

microcontroller. Arduino checks for this step with the application through the cloud only after which the vehicle is allowed in. The user should park the vehicle in the booked slot. IR sensor will detect the presence of the vehicle after parking in the slot and sent to the Node MCU. All the required statistics are sent to the cloud by the Node MCU which has a built in Wi-Fi module for future reference. Before the time up the user alerted with a message to vacate the slot. If not done the user is charged for the extra time consumed. This will be done when the user reaches the exit side of the gate which is detected by the IR sensor placed near the gate. After receiving the signal Arduino checks the time taken and the extra charge deduction process takes place. If no extra time is taken, then the vehicle is allowed to go out.

Android application:

Android app is designed using Android Studio which is software compatible with all the android devices and other hardware components. Using this there is no need for coding the designing tools and only main code is edited for the working of the design. This is best known and good working software at any condition for application development.

Android Studio with the latest version and built in features is available in Google for download. Along with that some libraries required for the application development is to be installed. Android Studio lists the required library for an application which developer can install easily. After installing next step is to create a folder for the app. The developer needs to enter application name of his choice and other details of supporting android version as per developer's requirement.

When the developer starts the project they need to select the main activity for coding and designing activity. These two-place important role in application development along with emulator. In the main activity the developer can find the main syntax of the code where he gets to add library which they can add as they mention new keyword in the code as well as the main block of the code. So, the developer only gets to write the main code inside public or private class that's one better feature by Android Studio. In the next side it has 2 part, one is design and other is text. The design part is for designing the slides with the required tools and the text part is the back-end code for the same design which is XML. The XML code is automatically updated by the system as we choose the tool with the adjustment. It contains the details of the tools such as dimension, ID, margin, colour and so on. To run the project, the software has emulator which has phone look like structure appears on screen after we click on run. The emulator works same as phone. It directs to our

application through chrome. This will be useful for the developer to know how the application going to display in the actual phone. And he can make changes if there is any correction. The developer also as an option to connect their phone through USB port but there will a chance of not supporting in some circumstance so better go for emulator until complete creating the application.

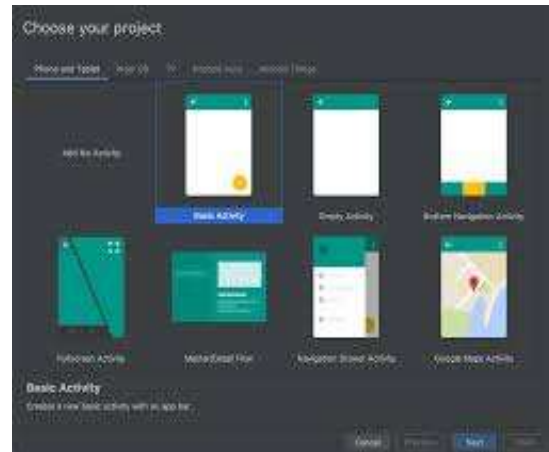


Figure 4. Activity choosing page

In our application we are going to design for booking of a parking lot with amount deduction. So first we are going to design the front page which will have a main heading and buttons. In an empty activity we will design along with the blue print beside which will show the dimensions and it's a option to keep it or not. But it's better to work with blue print on side so that we will be accurate with the adjustment. For the front page we are going to choose a main heading i.e. text field which will be constant in the application. For our application we are going to name it as "Welcome to Smart Parking". Next will be the main part our application as the whole process carried out through that. So, we are going to place 3 buttons in the front page which are Sign in, Register and Open gate. We are going to choose button tool for the same. Why button is because the booking process completes only after filling the pages which will be directed only by the buttons. So, on clicking the button it's going to direct us to next page. After placing 3 buttons in order we need to name the button with what it is going to be indicated as and also, we going to change the Id name if required. The Id will be named by default but for our convenience we will be naming with which we can recognise which button is it. This will be useful while coding when we choose the tool used by its Id. Now the button is renamed as SIGN IN and the same for the rest 2 buttons which are REGISTER and OPEN GATE. We can adjust the dimensions, margin, text size, colour, font of the tools. On click keyword is

used to rename the button and for the rest its available when we scroll down the editing tool bar. This is for the front page. Next page continues through the buttons. Register button directs to a page where user needs to fill some personal required details for booking purpose. These details are stored in the cloud and used whenever required. In the Register page the user needs to enter his name, vehicle number, mobile number, mail id, vehicle type to be more accurate and the last step is to create a password for a sign in process. This registration ends with DONE button and a notification, signing the successful completion of the process. The fields added in this are present in the tool bar such as the number field, Name field, password field and mail field. The user just needs to drag and arrange in the required manner. Next is Sign In page directed by the Sign In button, has 2 fields to the unique identification number and the password. This completes the Sign In step. The Sign in and Register option is added to have the user data and make the user a permanent member which will be even easy for the user to book the slot without need to spend much time. After completing the sign in step next is to choose the empty slot. There will 2 different pages for 2-wheeler and 4-wheeler as there is different section and gate for the vehicles in the parking area. The slot selection has 4 slots for now which will be in button form. To these slots a special feature is added to indicate the booked, empty and filled slots. This will be indicated with yellow, green and red slots respectively. After selecting the empty slot next step in the next page is to enter the required time for parking the vehicle. Next step is to deduct the money for the time entered by the developer as per the calculation. Amount deduction will be in terms of points as per now as the developer is not accessing the user account. This ends the booking process. A message or the mail will be sent to the user as a process completing notification with a unique code which is to be entered in the next step. So, the last button in the front page is OPEN GATE. The code sent to the user will be entered in this page which will be directed by the open gate button. This code will only work when the user reaches the parking area gate. So, the user needs to enter the code in this page when he reaches the entrance of the parking area which will also be fitted with the weight sensor. This completes the designing part.

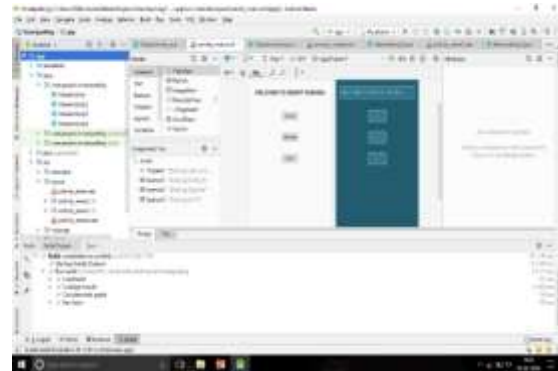


Figure 5. Design page



Fig 4.5 Front page and respective button page

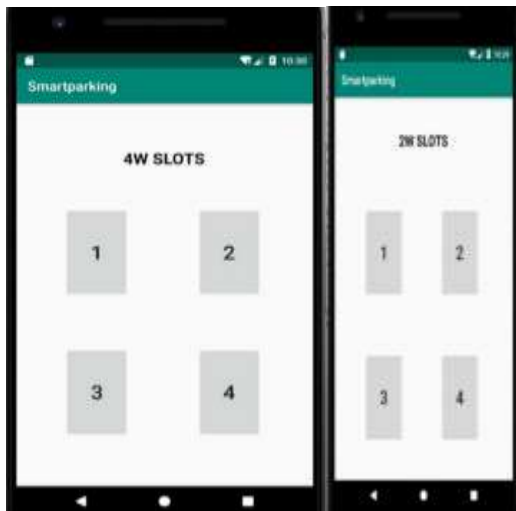


Figure 6. Slot choosing after SIGN IN

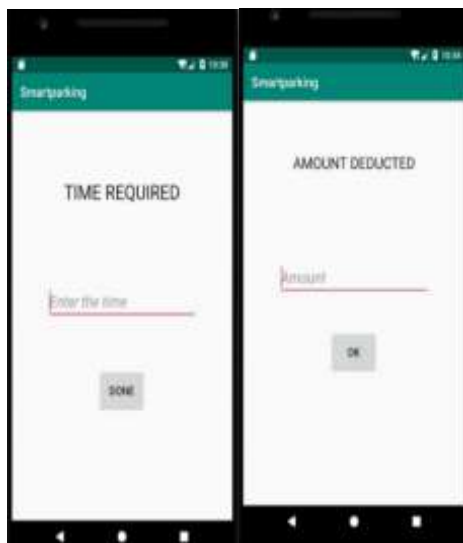


Figure 7. After slot selection pages

To make the design part work the developer needs to code for the same. The tools, and the fields used are to be recognized by the system and are made to function. So, in the main code the developer needs to mention the tools and the fields with the proper syntax and code. The tools like button code is mentioned below and also some main keywords like TOAST, INTENT, Log.i . Here Toast keyword is used to display a notification after each done or save button. Intent is used to direct a button from a current page to respective page. One more optional keyword used in the code that is Log.i which is to display a message in the terminal after clicking on a button. This is used while testing of button to check its working or not. This is about main terms used in the main code.

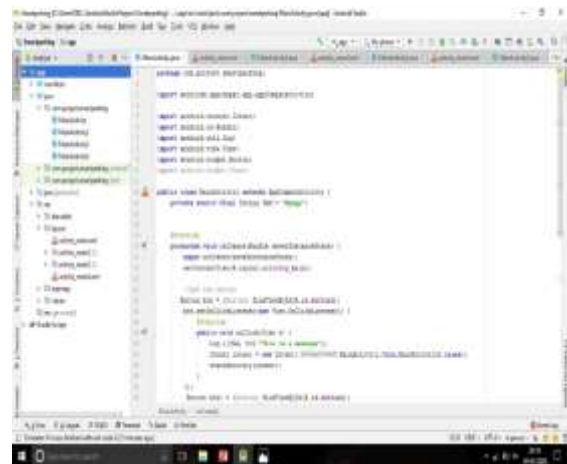


Figure 8. Coding page

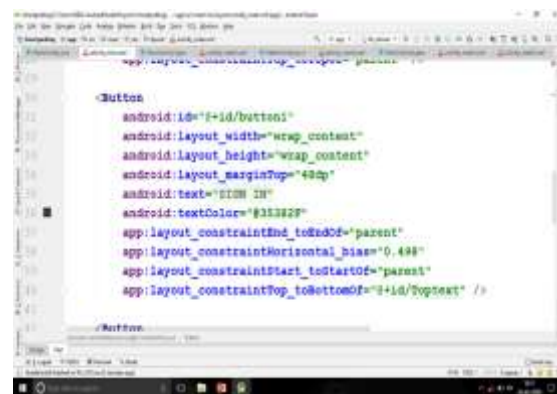


Figure 9. Back end XML code for button

After complete coding for each page the data from the required fields is made to store in the cloud. So, for this project the cloud used sap. The cloud is synched with the code as well as with the hardware component i.e., Node MCU a microcontroller which has a built in Wi-Fi module. The data stored from the application will be used by the microcontroller such as the open gate code likewise the slot status i.e., filled, booked or empty data will be updated in the application. This is about the Cloud functioning. And here completes the application part.

II. RESULT

All the components including sensors were neatly interfaced with the Arduino board. Figure 1 shows the project setup of the smart parking. The IR sensor detect the vehicle and it opens the gate by turning on the servo motor. The user will enter OTP near the gate which is sent during the booking of the slot. The user will park his vehicle in his slot. The slot will show the presence of vehicle in the android application.

III. CONCLUSION

Our smart parking reduces the congestion of vehicle in big cities. This will indirectly reduce the pollution in cities. User will find the secure place to park his vehicle. The traffic problem will be reduced for possible way. As you know we used internet of things to make smart parking easier this will reduce the man power. More features can be updated for the future in android application.

REFERENCES

- [1]. "IOT based smart parking", Abhirup Khanna University of Petroleum and Energy Studies (UPES) Dehradun, Uttarakhand, Rishi Anand University of Petroleum and Energy Studies (UPES) Dehradun, Uttarakhand
- [2]. David Angulo-Esguerra, Cesar Villate-Barrera, William Giral, Hugo Celedon Florez, A. Tatiana Zona-Ortiz, Felipe Díaz-Sánchez, "Parkurbike: An IoT-based system for bike parking occupation checking", Communications and Computing (COLCOM) 2017 IEEE Colombian Conference on, pp.1-5, 2017.
- [3]. IOT Analyst-ThingSpeak for IOT projects.
- [4]. Arduinio.cc.com
- [5]. Barton, J., J. Buckley, B. O'Flynn, S.C. O'Mathuna and J.P. Benson et al., 2007. The D-systems project-wireless sensor networks for car-park management. Proceedings of the 65th Vehicular Technology Conference, April 22-25, 2007, VTC2007-Spring, pp: 170-173.
- [6]. Muhammad Mahbubur Rashid, Abiodun Musa Aibinu, MD. Aatur Rahman Automatic Parking Management System and Parking Fee Collection Based on Number Plate Recognition International Journal of Machine Learning and Computing 2 2012.
- [7]. Abhirup Khanna, R. A. (2016). IoT based Smart Parking System. International Conference on Internet of Things and Applications (IOTA) (p. 5). Pune: IEEE.
- [8]. Deng, D. (2015). A Cloud-Based Smart-Parking System Based on Internet-of-Things Technologies. IEEE, 11.
- [9]. Mohit Patil, R. S. (2014). Smart Parking System Based On Reservation. International Journal of Scientific Engineering and Research (IJSER), 6.
- [10]. Vishwanath Y, A. D. (2016). Survey paper on Smart Parking System based on Internet of Things. International Journal of Recent Trends in Engineering & Research (IJRTER), 5.



**International Journal of Advances in
Engineering and Management**

ISSN: 2395-5252



IJAEM

Volume: 02

Issue: 01

DOI: 10.35629/5252

www.ijaem.net

Email id: ijaem.paper@gmail.com