

Table of Contents

[Keywords 4](#_Toc68037287)

[Introduction 5](#_Toc68037288)

[Aims and Objectives 6](#_Toc68037289)

[Aim 6](#_Toc68037290)

[Objectives 6](#_Toc68037291)

[Research 7](#_Toc68037292)

[System Design 7](#_Toc68037293)

[Circuit Design 8](#_Toc68037294)

[Implementation of the system 10](#_Toc68037295)

[Plan Project 13](#_Toc68037296)

[Risk Analysis 14](#_Toc68037297)

[Trouble shooting of the Project 15](#_Toc68037298)

[Problem Justification of the System 16](#_Toc68037299)

[Components used in the system: 17](#_Toc68037300)

[Tools and technologies 24](#_Toc68037301)

[Testing of the System 27](#_Toc68037302)

[Privacy and Ethics 28](#_Toc68037303)

[Personal Challenges 28](#_Toc68037304)

[Future Work 29](#_Toc68037305)

[Conclusion 30](#_Toc68037306)

[References 31](#_Toc68037307)

[Appendix 32](#_Toc68037308)

[The sketch 32](#_Toc68037309)

[Firebase setup 40](#_Toc68037310)

[Android App 43](#_Toc68037311)

Table of Figures

[Figure 1: Ultrasonic sensors configuration 8](file:///C:\Users\Aavash\Downloads\SmartParkingSystemIOT-Group-Project.docx#_Toc68037312)

[Figure 2: Gate Configuration 9](#_Toc68037313)

[Figure 3: Working Mechanism of the system 10](#_Toc68037314)

[Figure 4: Smart parking gate 11](#_Toc68037315)

[Figure 5: Smart Parking slots 11](#_Toc68037316)

[Figure 6:Gate 12](#_Toc68037317)

[Figure 7: Task of the System 13](#_Toc68037318)

[Figure 8: Gantt Chart 13](#_Toc68037319)

[Figure 9: Before using Smart Parking Application 16](#_Toc68037320)

[Figure 10: After using Smart Parking Application 16](#_Toc68037321)

[Figure 11: Components Used in the system 17](#_Toc68037322)

[Figure 12: NodeMCU esp8266 18](#_Toc68037323)

[Figure 13: Ultrasonic Sensor 19](#_Toc68037324)

[Figure 14: Jumper Wire/ Rainbow Wire 20](#_Toc68037325)

[Figure 15: Double Channel Relay 21](#_Toc68037326)

[Figure 16: Wiper Motor 22](#_Toc68037327)

[Figure 17: DC Power Supply 23](#_Toc68037328)

[Figure 18: Tools used in this system 24](#_Toc68037329)

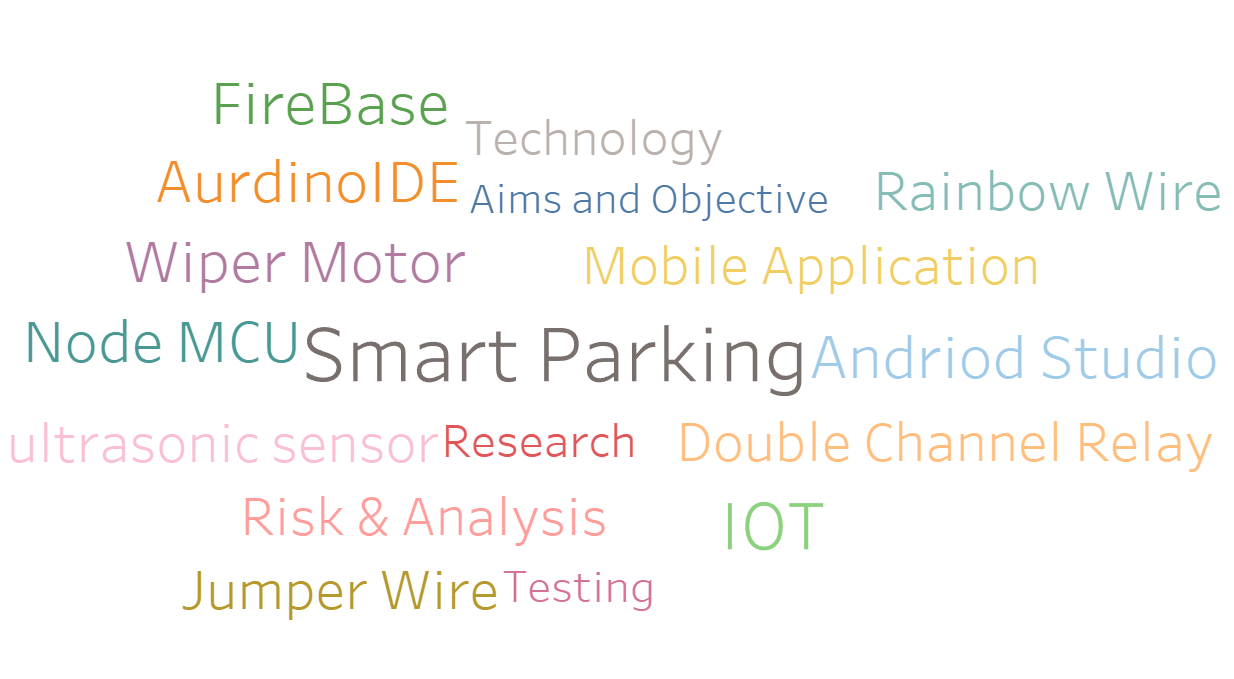
[Figure 19: Arduino IDE 25](#_Toc68037330)

[Figure 20: Firebase 25](#_Toc68037331)

[Figure 21: Android Studio 26](#_Toc68037332)

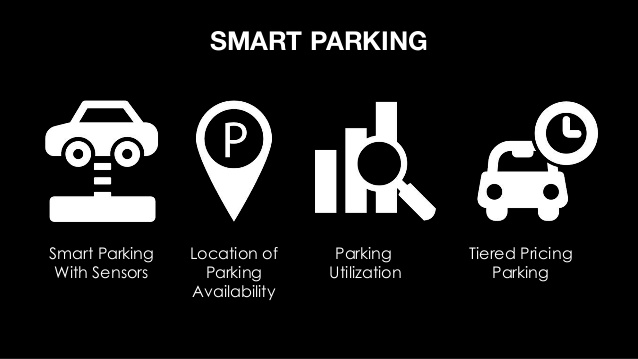
[Figure 22: GitHub 26](#_Toc68037333)

# Keywords

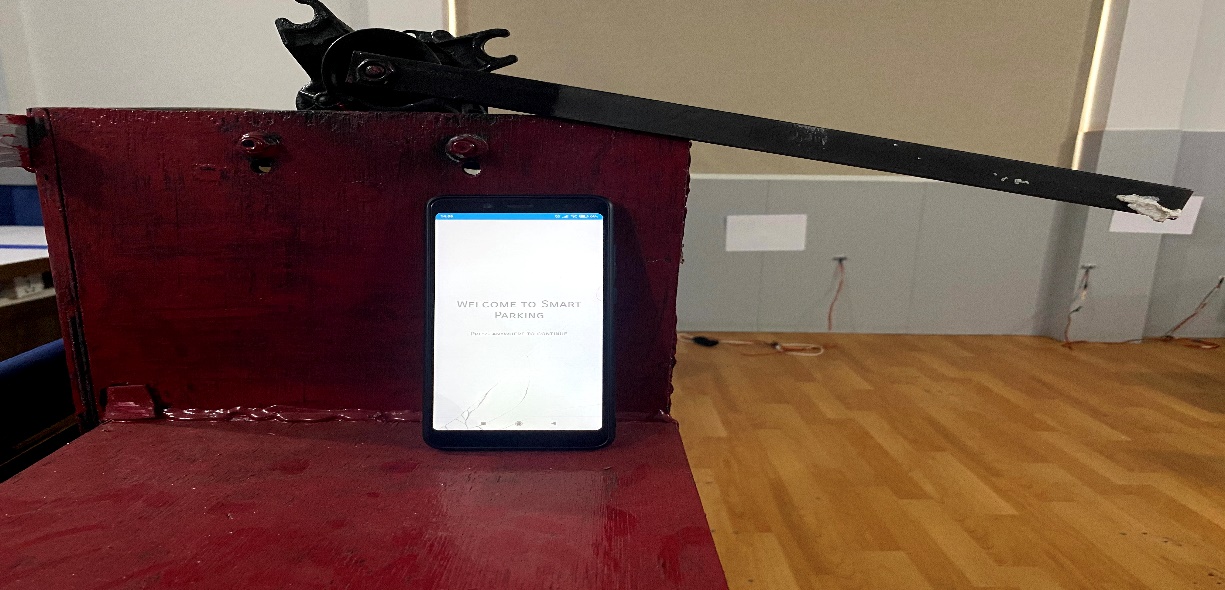


# Introduction

According to the present situation, it is hard to find a parking spot in rush hour in crowded places such as shopping centres, hospitals, and exhibits. The challenge arises from not understanding where the open spots will be at the moment. As a result, Smart Parking is a way to decrease traffic congestion, and save people's time by assisting them in locating a parking space at the right time.



Smart Parking is an Internet of Things (***IoT***)-based technology that detects open parking spaces and lets user book and enter/exit the parking lot seamlessly. This program senses the presence of a vehicle using an ultrasonic sensor (whether the parking slot is occupied or not). The condition (occupied/empty) is shown on the mobile device dashboard depending on the parking slot occupancy. In real time, sensors, and devices such as microcontrollers (***NodeMCU 8266***) are installed in parking spaces, transmitting data on the occupancy status, and car drivers can search for open parking spots using their mobile phones. As a result, the driver will know where there is an available parking space in less time, reducing the energy consumption. The sensor posts the parking slot occupancy status to a mobile application dashboard. The Kotlin based mobile application is built using Firebase to connect to the backend system. Arduino IDE is used to detect the values obtained from the ultrasonic sensors and is sent to the real time database (***Firebase***) to transmit the data on the occupancy status.



# Aims and Objectives

Aim  
The aim of the project is to develop an intelligent, user friendly automated parking system which reduces the manpower and increases user convenience through appropriate IOT based solutions.

## Objectives

* To implement the system using sensors that can detect parking lot status in real time.
* To establish communication between multiple sensors, devices, database and application.
* To develop mobile application that can assist user in the process of parking.
* To monitor the condition of parking lot through application.
* To develop system that can book parking slot if available and reserve it for a certain time.
* To develop a system that can generate unique code which can be used in parking lot gates.
* To make a system that can open entry and exit gate if correct code is entered.
* To develop a system that can calculate timestamp and deduct amount from dummy payment system.

# Research

## System Design

The specification for incorporating the Smart Parking Scheme has been developed in such a way that it assists in the seamless and efficient execution of parking process. The devices were chosen after carefully analysing their feasibility in the project . Devices like ultrasonic sensors, NodeMCU8266, double channel relay, wiper motor, DC power supply, Tablet, Rainbow Wires have been used for making the prototype of the system.

Mentioned sensors have been used for the smart parking system to function properly and these sensors are used for the following purposes:

* Ultrasonic Sensor has been used for detecting the slot occupancy status.
* NodeMCU8266 has been used for transmitting the data obtained from the ultrasonic sensor to the Firebase.
* Double-Channel Relay has been used to trigger the signal of the wiper motor.
* Wiper Motor has been used to control the opening and closing of the prototype of the gate.
* DC Power Supply has been used as a power supply source for the wiper motor.
* Tablet has been used to run the mobile application.
* Rainbow wires has been used for the configuration of the prototype of the system.

The working prototype of the project can be found in:

**Github:**

Sensors:<https://github.com/aavashneupane/IOT-SmartParking>

Client app: <https://github.com/aavashneupane/SmartParkingClient>

Gate app: <https://github.com/aavashneupane/SmartParking>

**Youtube:**

[**https://www.youtube.com/watch?v=1fDKVc2\_KAs**](https://www.youtube.com/watch?v=1fDKVc2_KAs)

## Diagram, schematic Description automatically generatedCircuit Design

Figure 1: Ultrasonic sensors configuration

Two ultrasonic sensors have been connected to the ***NodeMCU 8266 Module.*** VCC and GND pins of the ultrasonic sensors has been connected to the VIN and G pins of the NodeMCU. Trigger and Echo of both ultrasonic sensors have been connected to the digital pins (D1, D2, D5, D6) of the NodeMCU.

Diagram

Description automatically generated

Figure 2: Gate Configuration

This is the diagrammatic illustration of the prototype of the gate. VCC of the Double Channel Relay is connected to the VU of the NodeMCU and GND of the relay is connected to the G pins of the NodeMCU. The NO and NC of relay are connected to battery and COM of relay are connected to positive and negative terminal of motor.

# Implementation of the system

For demo purposes, the system was implemented in three slots. However, it can be scaled to a large parking space as well.  
For each parking slot, the detection of vehicles was done by two ultrasonic sensors which was attached to a NodeMCU. If only one sensor had been used, then the sensor could identify any object as vehicles and send inaccurate data. So, in order to overcome this, two sensors were implemented in such a way that data was sent to firebase only when both sensors detected obstacles after reaching a certain threshold.  
If any vehicle was detected, then the NodeMCU from the slot sends occupied value in database and user cannot book a slot. However, if there is vacant space available then user can book a slot and reserve it for five minutes. During the five minutes the booking is kept closed for that slot and user needs to come to the space within that time otherwise the booking would be reset.  
After successfully booking a slot, the user receives a notification containing a unique five-digit code which can be used in gates for entering and exiting. After the user enters a correct code in the gate, the gate opens and closes automatically. This is handled by establishing communication between NodeMCU, database, double channel relay and wiper motor.  
After the user enters and parks the vehicle in the slot, a timestamp is recorded which marks the start of parking time. In order to exit the parking lot, similar process is repeated where user goes to exit gate and enters the same five-digit code which was obtained while booking. If the code is correct, then the gate is opened, and another timestamp is recorded which marks the end of parking time. After that, time difference is calculated, and certain value is deducted automatically from user account.

Diagram

Description automatically generated

Figure 3: Working Mechanism of the system



Figure 4: Smart parking gate

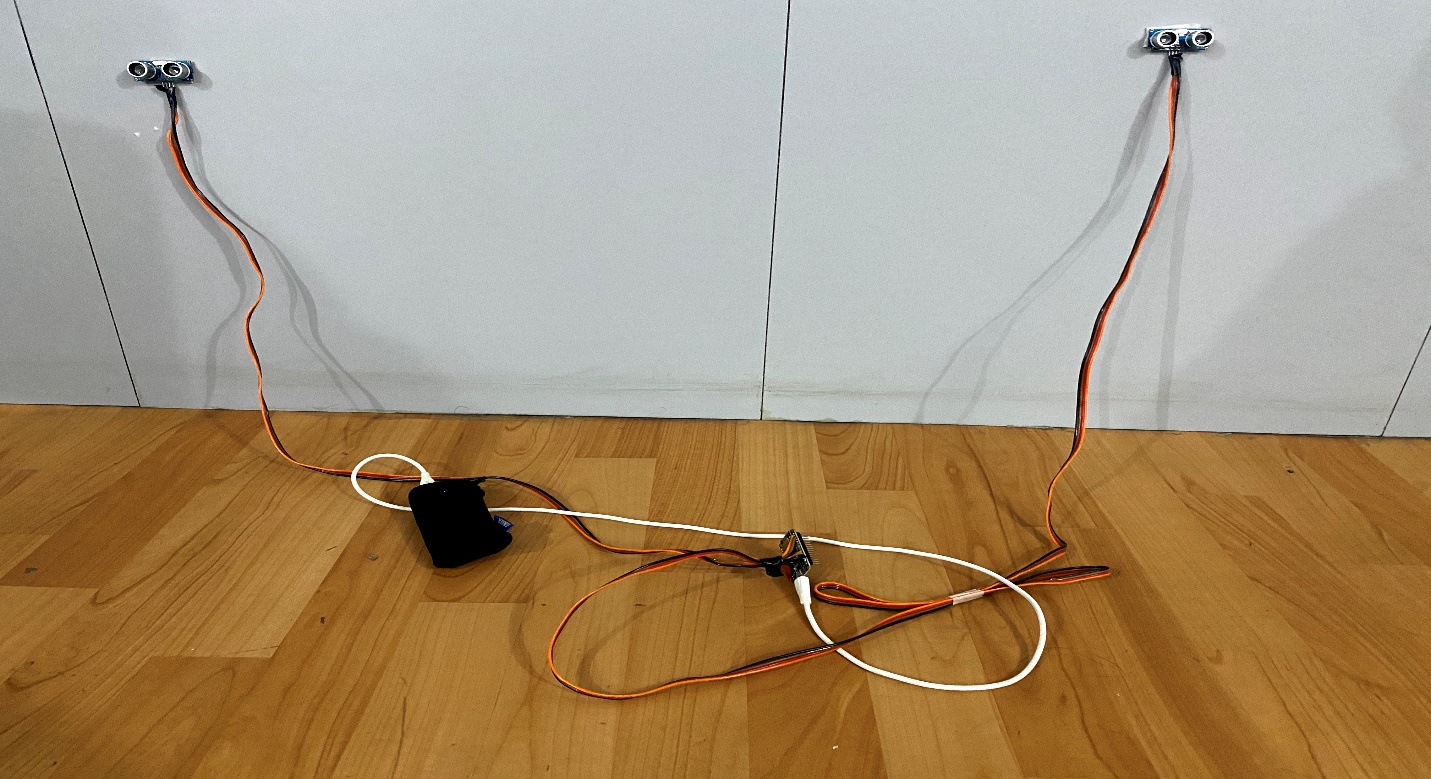


Figure 5: Smart Parking slots



Figure 6:Gate

# Plan Project

The project has been carried out by following a proper plan starting from project initiation and planning to handover and finalizing the document.

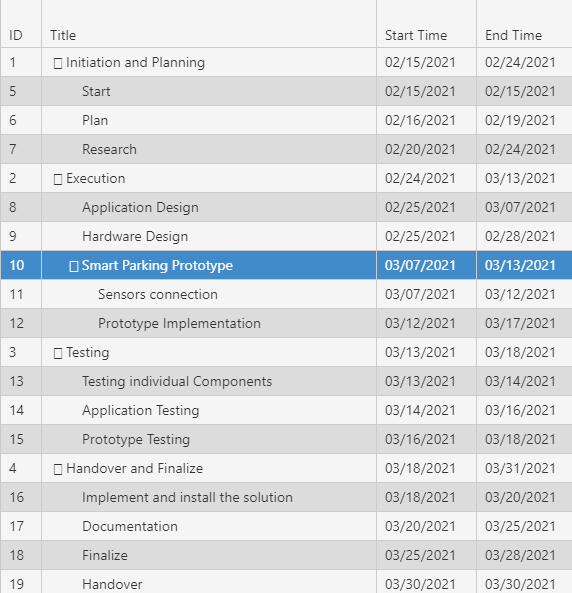


Figure 7: Task of the System

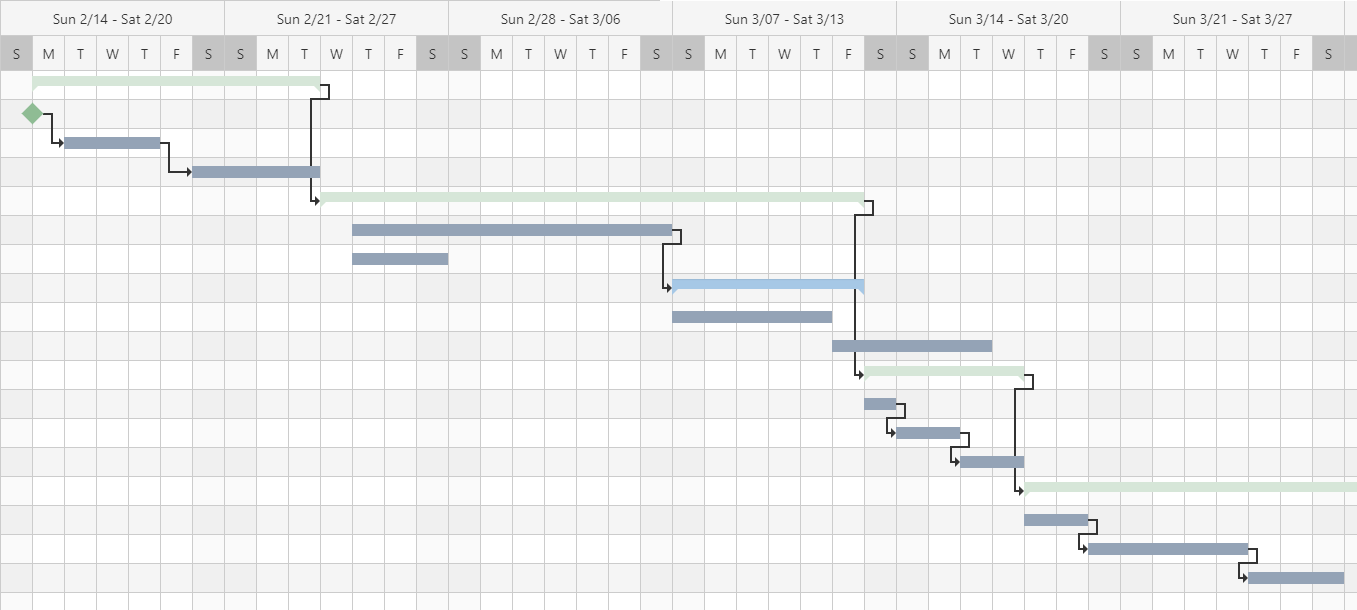
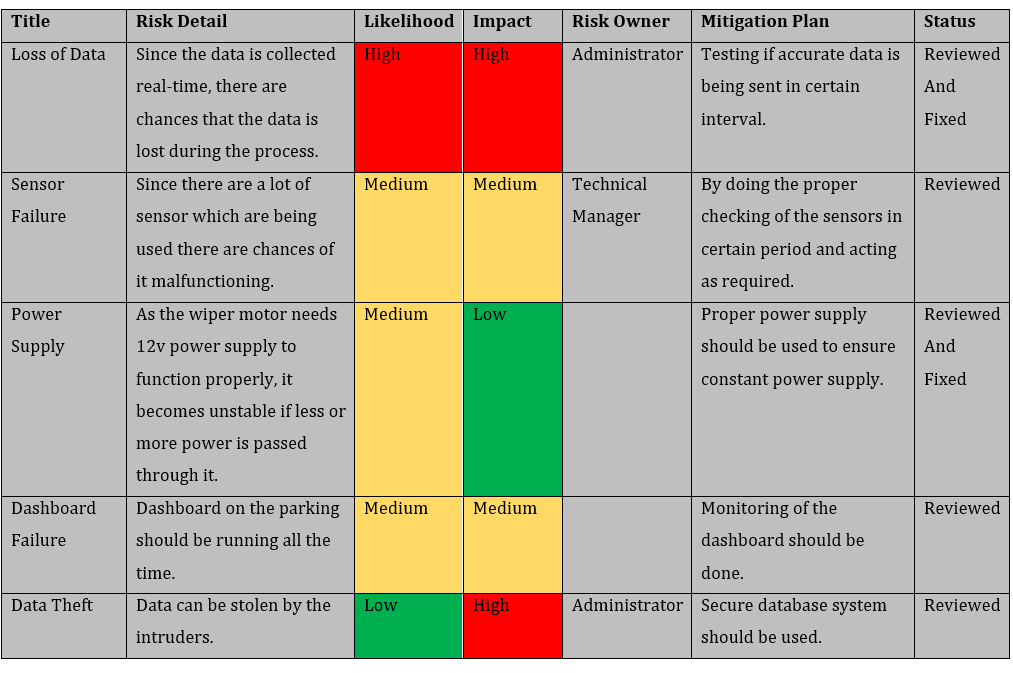


Figure 8: Gantt Chart

# Risk Analysis

Risk analysis has been carried out in order to overcome certain challenges while doing project. It has been designed in such a way that it helps to foresee certain risks, see its impact while developing an appropriate solution.



## Trouble shooting of the Project

**Q. Parking is full even if the slots are empty**

* Check if all the sensors are working properly and is sending correct information and check internet connection of NodeMCU and mobile device.

**Q. Gate not opening**

* Ensure that the motor is receiving proper voltage of the power supply i.e 12v and wires are connected in correct order.

**Q. Code entered error**

* Check internet connection and enter code according to booked slot.

# Problem Justification of the System

Diagram

Description automatically generated

Figure 9: Before using Smart Parking Application

The first story board illustrates the parking problems faced by the person. He could not park his vehicle in time due to full parking spaces and he was very upset that he could not attend his meeting and wished if new innovative technology could overcome current parking problems.



Figure 10: After using Smart Parking Application

The second storyboard illustrates the person attending the meeting and he was thinking he will be late again but he remembered that he had installed the mobile application where the users can prebook their parking slots by observing the occupancy status of the parking area. He is glad that smart parking application have replaced the previous parking problems and thanks the system.

# Components used in the system:

All the components used in the project are described below.

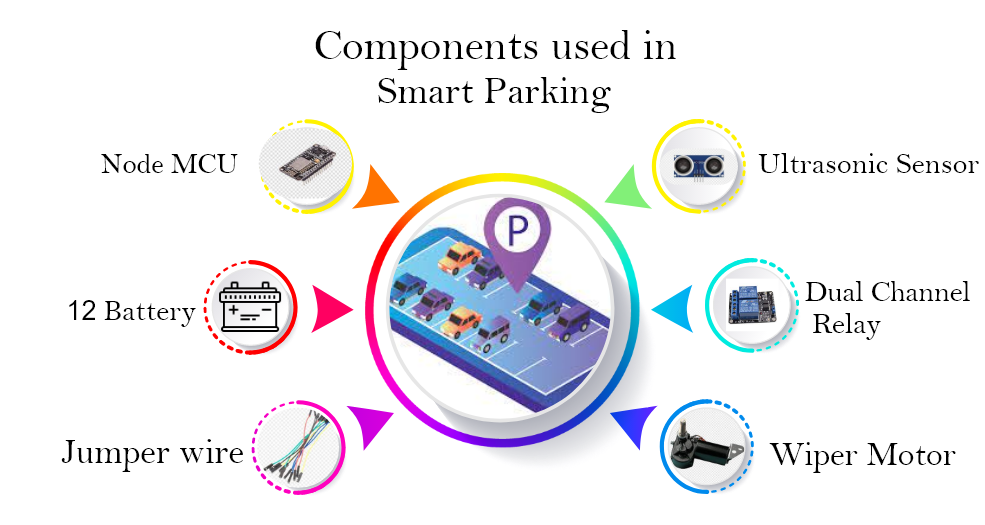


Figure 11: Components Used in the system

**Node MCU:**

The NodeMCU is a microcontroller panel based on ESP8266. It includes the firmware which runs on the WIFI capability. It is the combination of the GPIO, PWM, IIC,1-Wire, and ADC all together into one board. NodeMCU has been favored against other microcontrollers in this project because of its Wi-Fi capability and support for multiple digital pins.

[(*NodeMCU ESP8266*)](#Node)

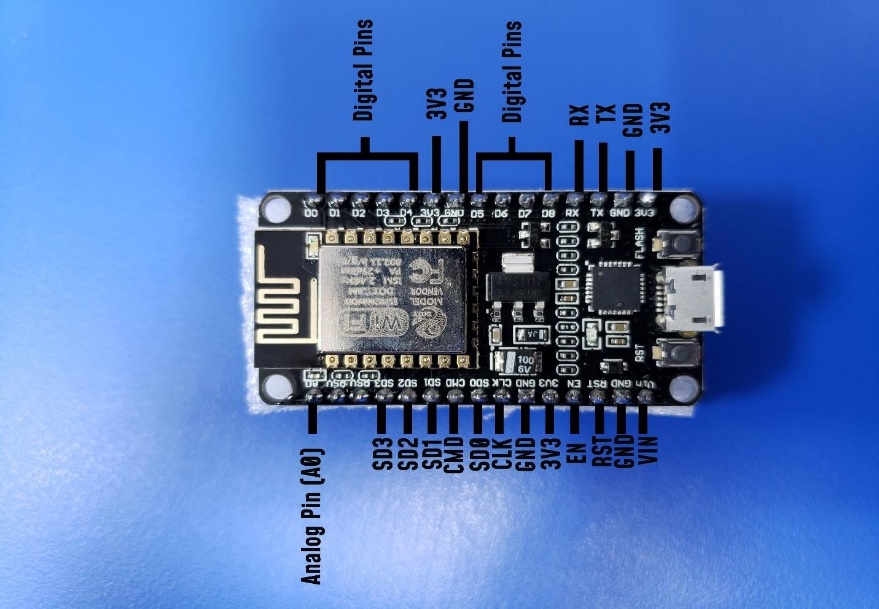


Figure 12: NodeMCU esp8266

## 

**Ultrasonic Sensor:**

An ultrasonic sensor detects the distance between two objects. It determines distance by emitting ultrasound and receiving the wave reflected by the object. Ultrasonic sensors have been used in the project to detect the presence of vehicle in parking lot.

[(](#ultra)*[What Is Ultrasonic Sensor](#ultra)*[2021)](#ultra)

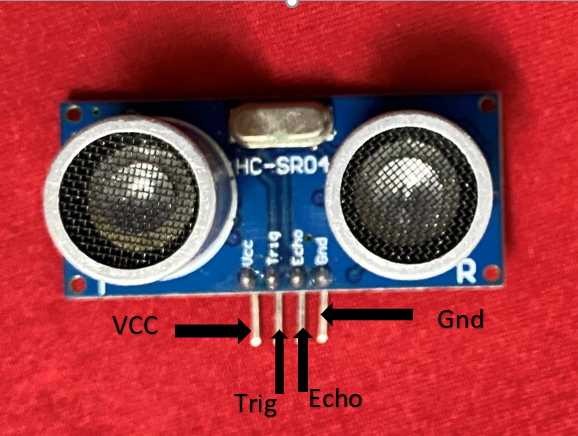


Figure 13: Ultrasonic Sensor

**Jumper Wire:**

A jumper wire is a conductor wire that connects two points in a circuit to pass electrical signals. The wires may be used to either change circuits or diagnose problems with them. Jump wires have been used to make contacts between both the main microcontroller and other equipment such as sensors and relay.

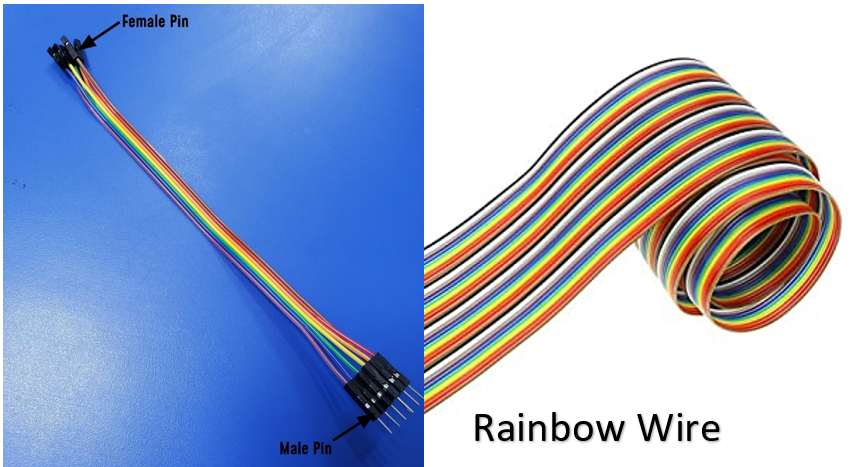


Figure 14: Jumper Wire/ Rainbow Wire

**Double-Channel Relay:**

A relay is an **electromagnetic switch** that is operated by a relatively small current that can control much larger current. Double channel relay has been used to switch voltage to operate the wiper motor in clockwise and anticlockwise direction according to signal sent from NodeMCU.

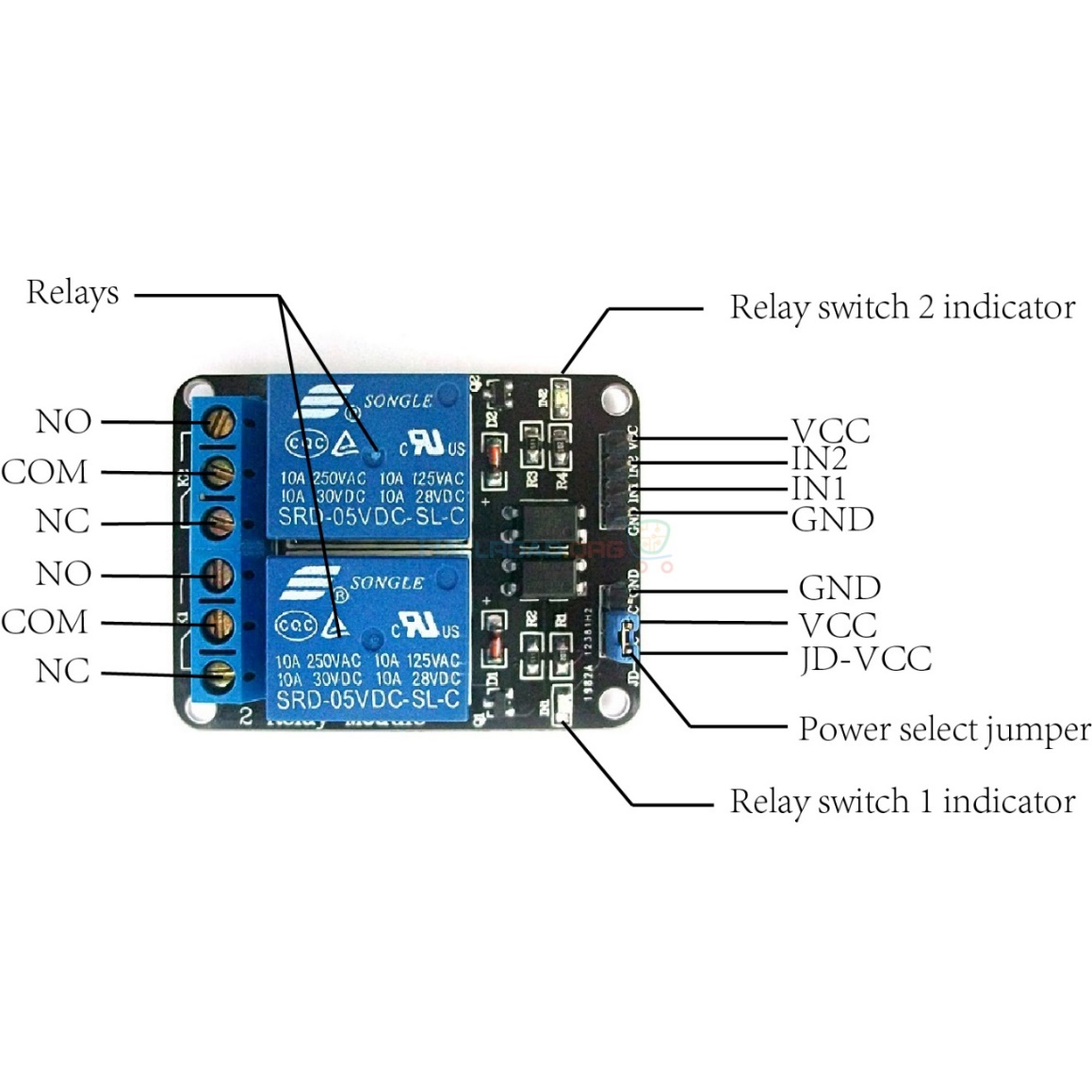


Figure 15: Double Channel Relay

## 

**Motor:**

Servo motors are also used to adjust the location of wing flaps in radio control airplanes and to spin the wheels in radio control vehicles. A servo's output shaft does not spin easily like a DC motor's shaft, but rather is made to seek a certain angular direction under electronic power. A servo motor is essentially a DC motor with a shaft location sensor and a feedback circuit built in.



Figure 16: Wiper Motor

**DC Battery:**

Six single cells are connected in series to provide a fully charged peak voltage of 12.6 volts in a twelve-volt battery. A standard 12-volt battery used in an RV or marine vessel has a 125 AH rating, which means it can have 10 amps of current for 12.5 hours or 20 amps for 6.25 hours.



Figure 17: DC Power Supply

# Tools and technologies

Various tools and technologies used in the project are described below.



Figure 18: Tools used in this system

**Arduino IDE:**

The Arduino Development Environment (IDE) is a cross-platform programming framework in C and C++ functions. It is used to write and upload programs to Arduino-compatible boards, as well as other manufacturer development boards with the support of third-party cores. In Arduino IDE all the connection of the components and code are written along with the testing of the output of the sensors. ([Aqeel 2021](#aa))



Figure 19: Arduino IDE

**Firebase**

Firebase is a platform created by google to create mobile as well as web applications. that offers a range of tools and resources to help developers create high-quality applications. Firebase real time database has been used to communicate with mobile application and NodeMCU. It has been used because the system needed real time status of the surrounding in order to operate smoothly. ([firebase](#fb))

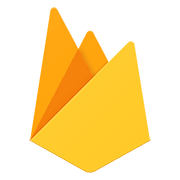


Figure 20: Firebase

**Android Studio:**

Based on JetBrains IntelliJ IDEA software and developed exclusively for Android production, android Studio is the official optimized development environment for Google's Android operating system. We have designed and created the application for our smart parking which were used for booking, going in and coming out from gate of smart parking. ([Android Studio](#AS))



Figure 21: Android Studio

**GitHub:**

For app developers, GitHub is a web-based version management and sharing site. GitHub was for collaboration for system development. ([GitHub](#gg))

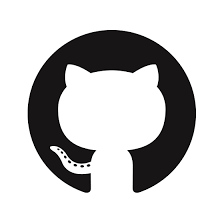
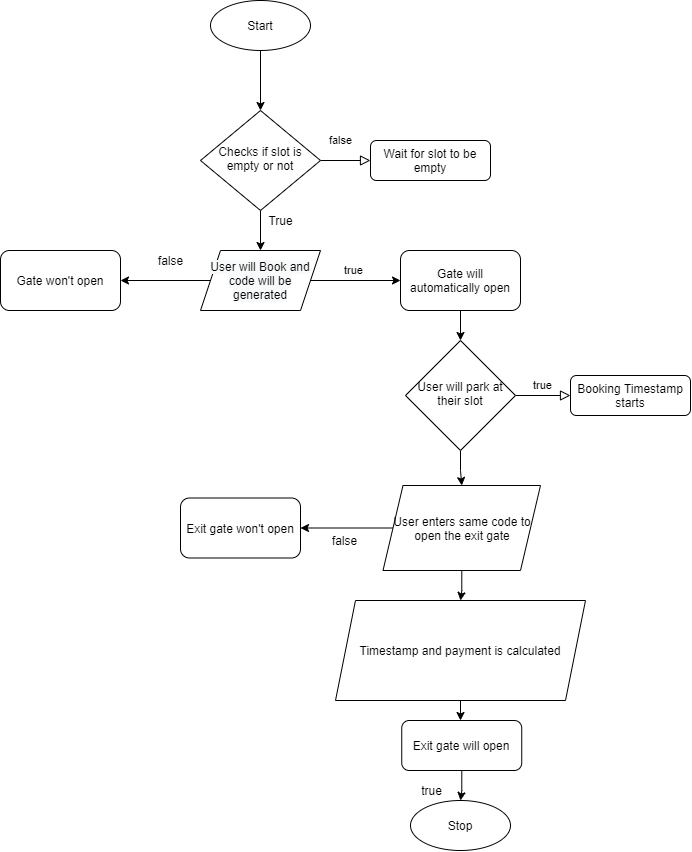


Figure 22: GitHub

# Testing of the System

Testing is the important part which is done after the project is completed. By doing the test we can ensure that the projects functional and it works according to our expectation. The system used four components which are NodeMCU, Ultrasonic sensor, Double channel relay and Wiper Motor. The sensors and other electronic devices were tested separately. Each of them where first developed separately and tested to insure there were not any fault while it functioned. Firstly, the ultrasonic sensor was tested as we used six of them in total. The double channel relay was tested to ensure that the gate was opening and closing properly. The wiper motor is used for opening and closing the gate which is connected to the relay for clockwise and anti-clockwise movement. There were few problems while doing the test but after multiple testing the problems were solved thus, the testing was passed. 

# Privacy and Ethics

Privacy and Ethics are one of the important things to consider while developing any project. An ethical approach to data privacy makes sure that there is trust and security in the system. Our project is based on collection of the data from user as well as the sensors. If these data are handled carelessly the system can be breached easily. Therefore, to ensure data privacy and security the data was stored in secure database. Also, a random token generator is implemented to ensure that only the person with the authorization can access the system. As the ethical consideration for the data collection and manipulation only the necessary data which are needed are collected and demonstrated.

# Personal Challenges

Various challenges were faced during the development of the project. Some are discussed below.  
There were various complications while establishing communication between different sensors. The ultrasonic sensors didn’t communicate properly via a single NodeMCU. In order to resolve this, each slot was given two ultrasonic sensors and a NodeMCU which then sent accurate data to the database.   
The main challenge was controlling the speed of wiper motor which was used as gate for in and out. This was because if the motor was supplied with power, it rotated 360 degree. Unlike servo motor, it didn’t come to its initial position after supplying power. To overcome this, double channel relay was used in the motor to send the signal as front and reverse in certain interval of time according to value received in NodeMCU through firebase.

# Future Work

This application is a first step toward finding an appropriate approach to an everyday problem. This project can be expanded in several ways:

* To have a centralized management mechanism that ensures only authenticated information is transmitted to the Client, thus addressing security concerns.
* More analysis can be conducted using the parking history details, helping the user to get feedback or advice on parking spaces and their availability patterns.
* The obtained analysis can be used by the user when reserving a parking spot or renting a space to calculate the price for the parking space.
* Implementing a QR-code system so user does not have to write the codes by themselves.
* Implementation of a real payment system.

# Conclusion

Smart Parking System is a solution to the existing traffic congestion, to reduce drivers’ frustration providing information about the occupancy status of the parking spaces. The project development went smoothly while teaching us many best practises in programming using the current trending technologies like Kotlin, Arduino, Firebase. We could see that all the initial requirements of the project were achieved.

The mobile application is user friendly that can assist user to easily find the status(vacant/occupied) of the parking space. Starting from coming up with project idea, understanding the requirements and choosing the best technologies for the implementation, all this assisted in developing a working prototype of a smart parking system that can be implemented in a closed parking lot.

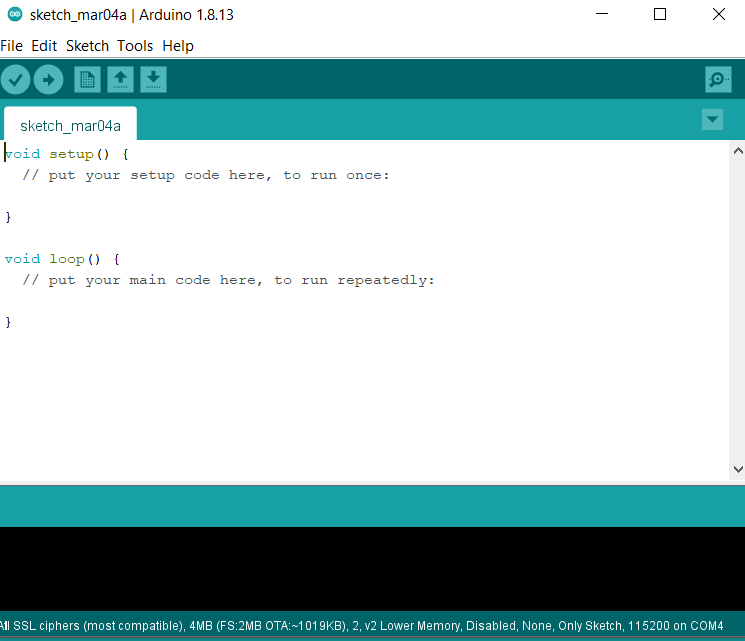
# References

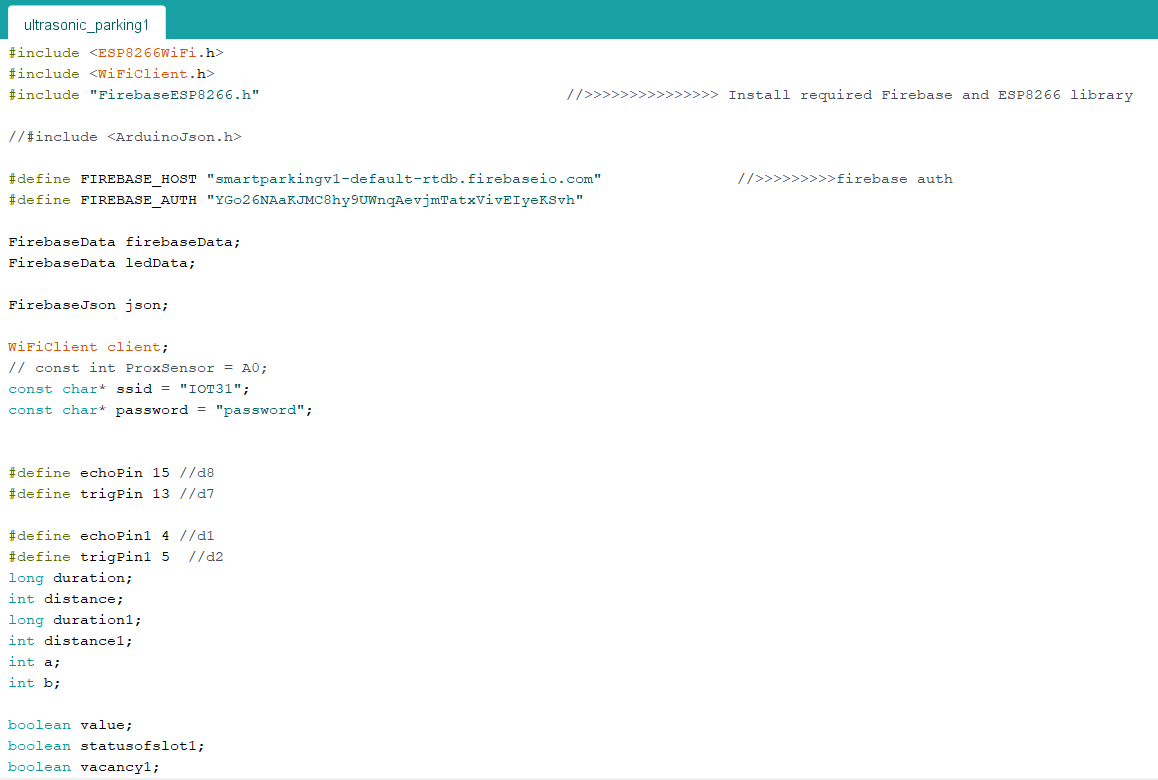
* Cse.unl.edu. 2021. [online] Available at: <http://cse.unl.edu/~byrav/INFOCOM2011/workshops/papers/p701-wang.pdf> [Accessed 27 March 2021].
* *Nodemcu ESP8266 Pinout, Specifications, Features & Datasheet* (2021) available from <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet> [25 March 2021]
* What Is Ultrasonic Sensor (2021) available from <https://www.sonotec.com/en/column/ultrasonic.html> [27 March 2021]
* Aqeel, A. (2021) *Introduction To Arduino IDE - The Engineering Projects* [online] available from <https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html> [27 March 2021]
* *What Is Firebase?* (2021) available from <https://www.educative.io/edpresso/what-is-firebase> [27 March 2021]
* *What Does GITHUB Mean?* (2021) available from <https://www.definitions.net/definition/GITHUB> [27 March 2021]
* *What Is Android Studio? - Definition from Whatis.Com* (2021) available from <https://searchmobilecomputing.techtarget.com/definition/Android-Studio> [27 March 2021]
* Market Business News. 2021. *What is risk analysis? Definition and meaning - Market Business News*. [online] Available at: <https://marketbusinessnews.com/financial-glossary/risk-analysis-definition-meaning/> [Accessed 23 March 2021].

# Appendix

## The sketch

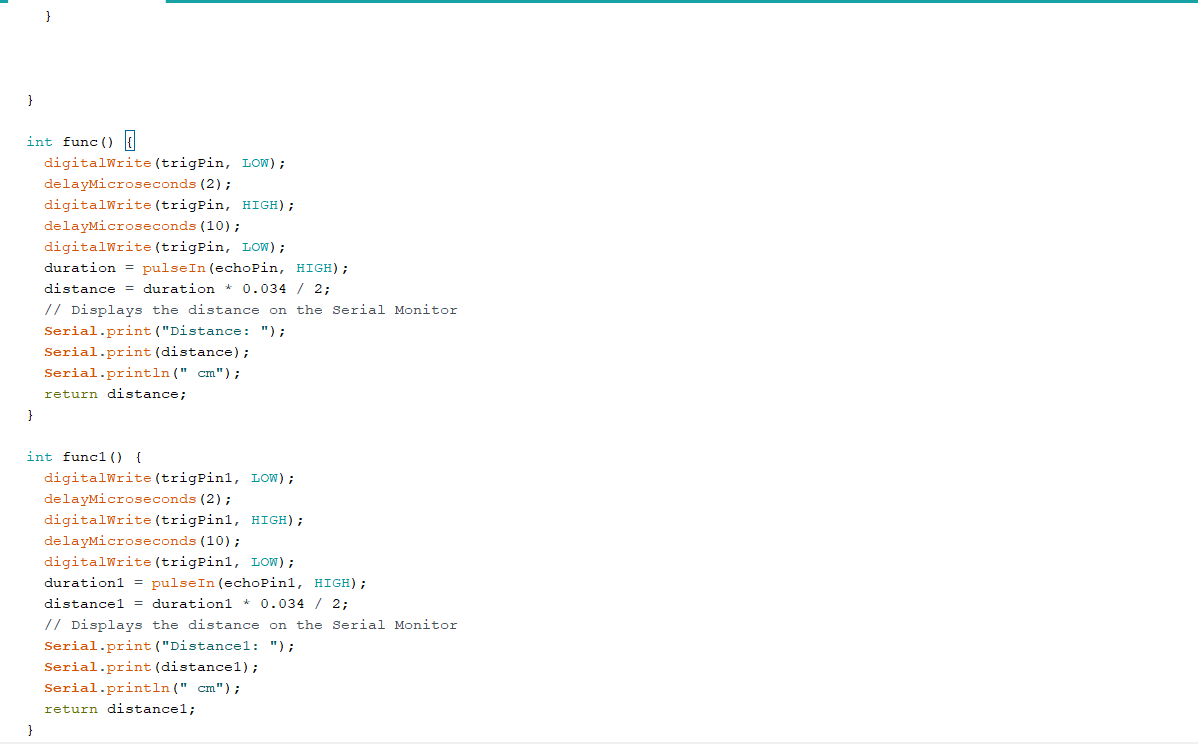
Arduino IDE was used as a coding environment for the system.







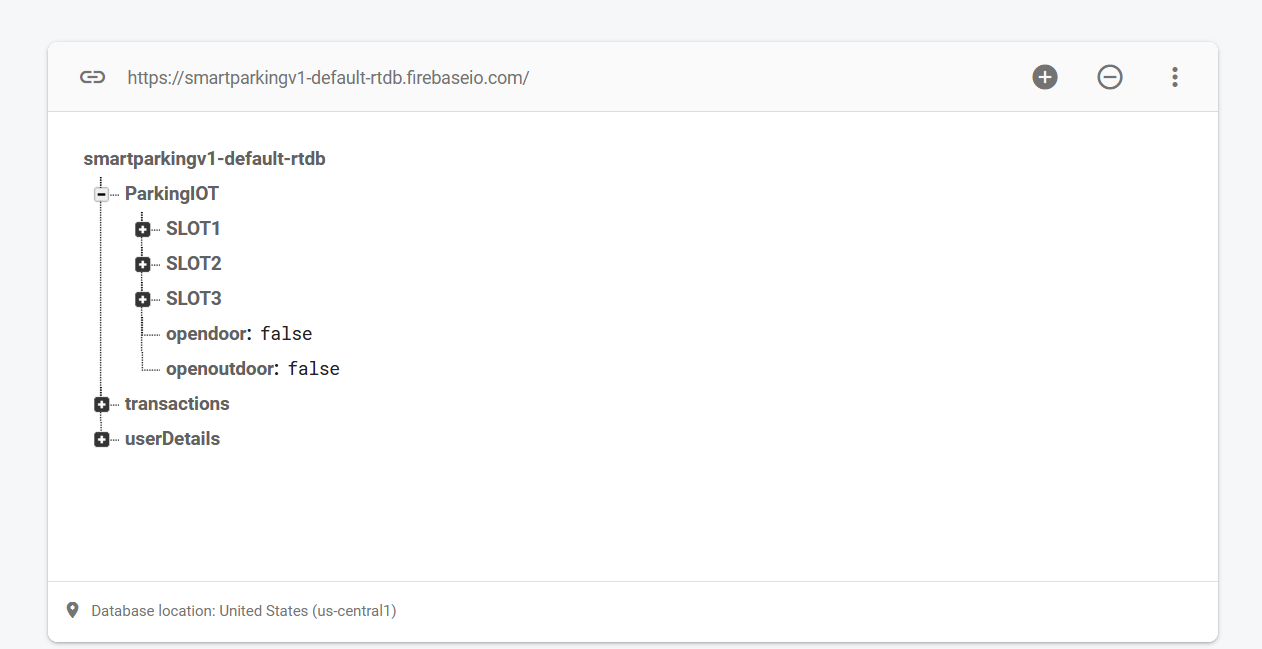


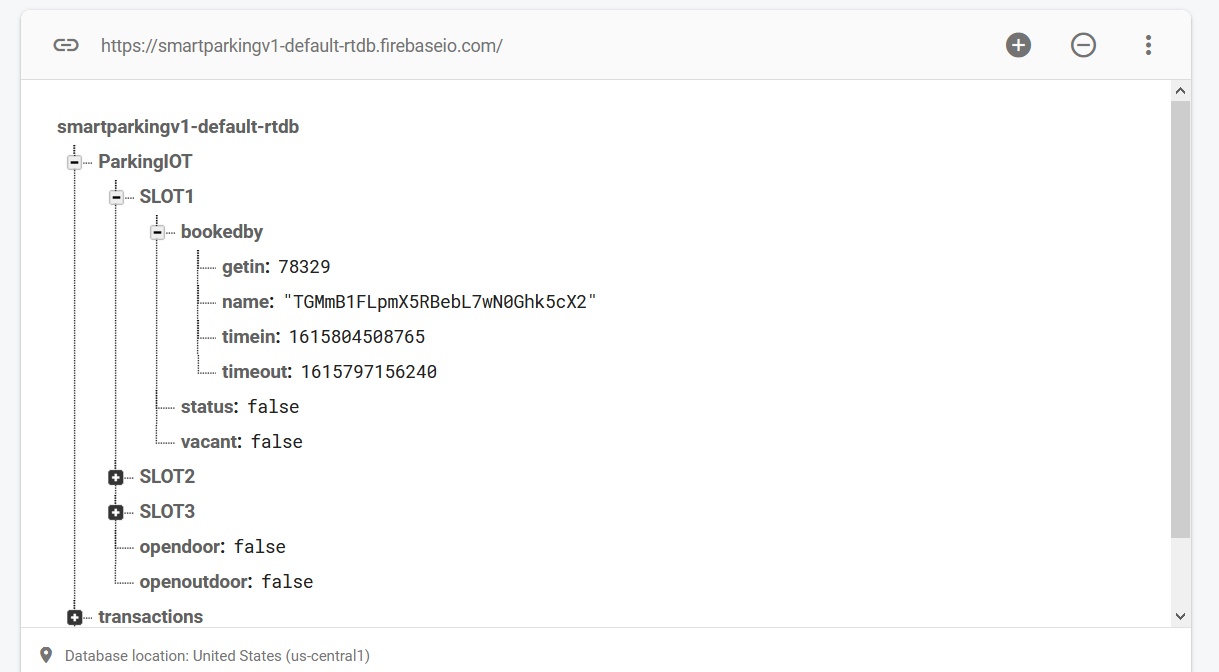


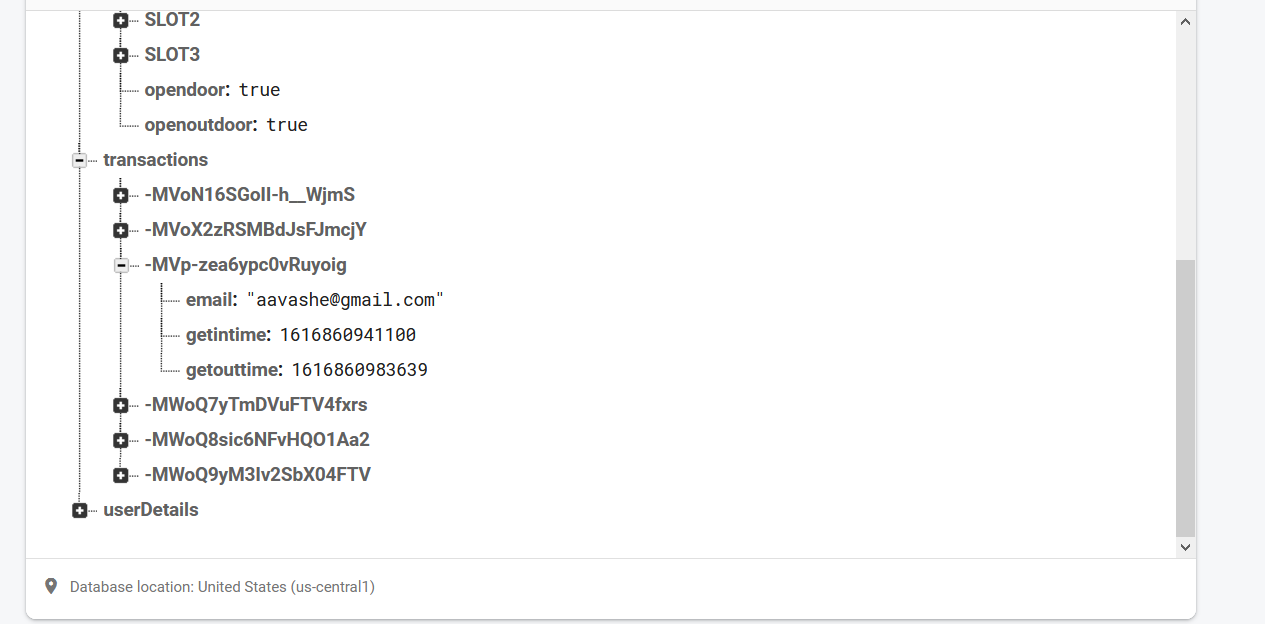


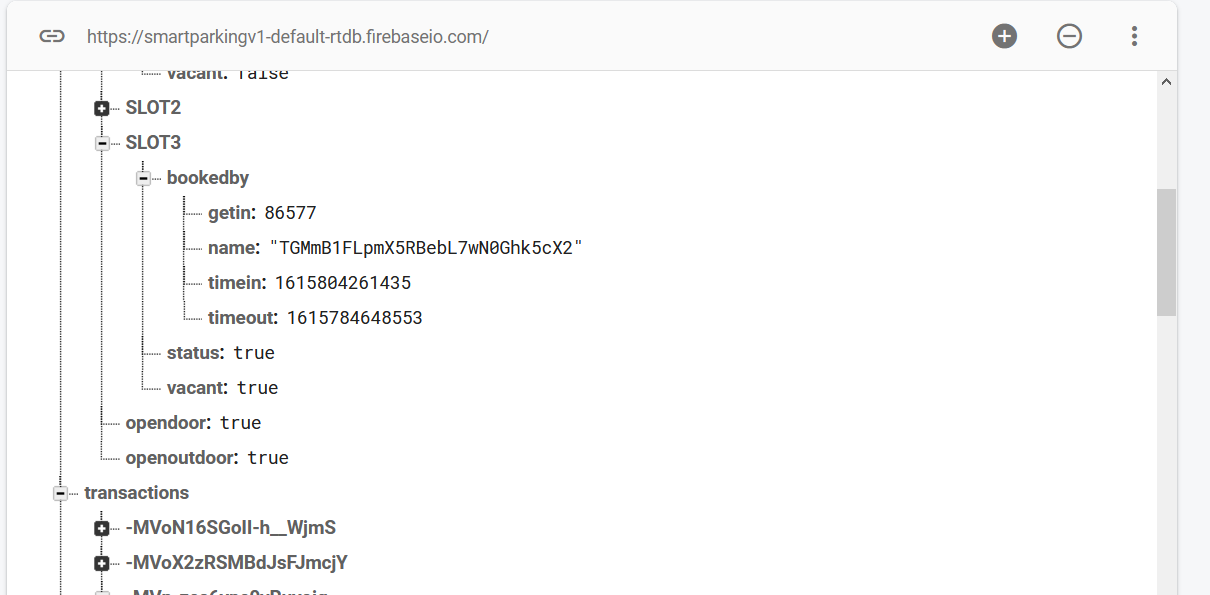
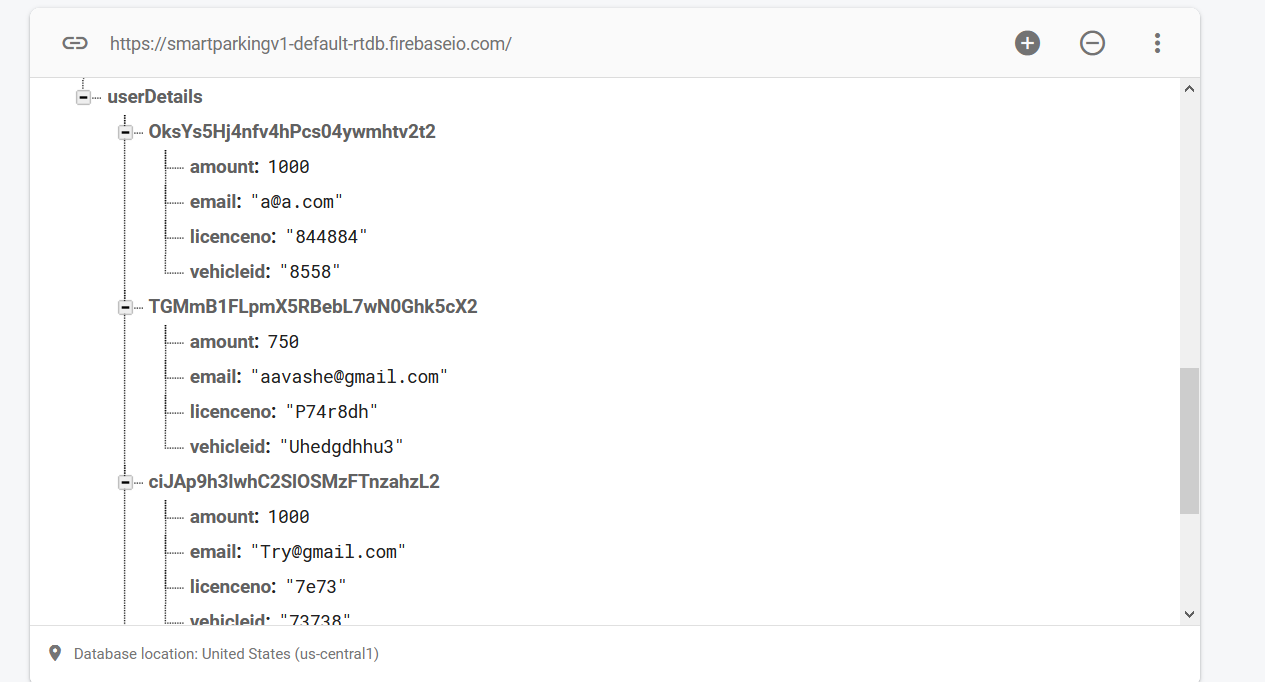


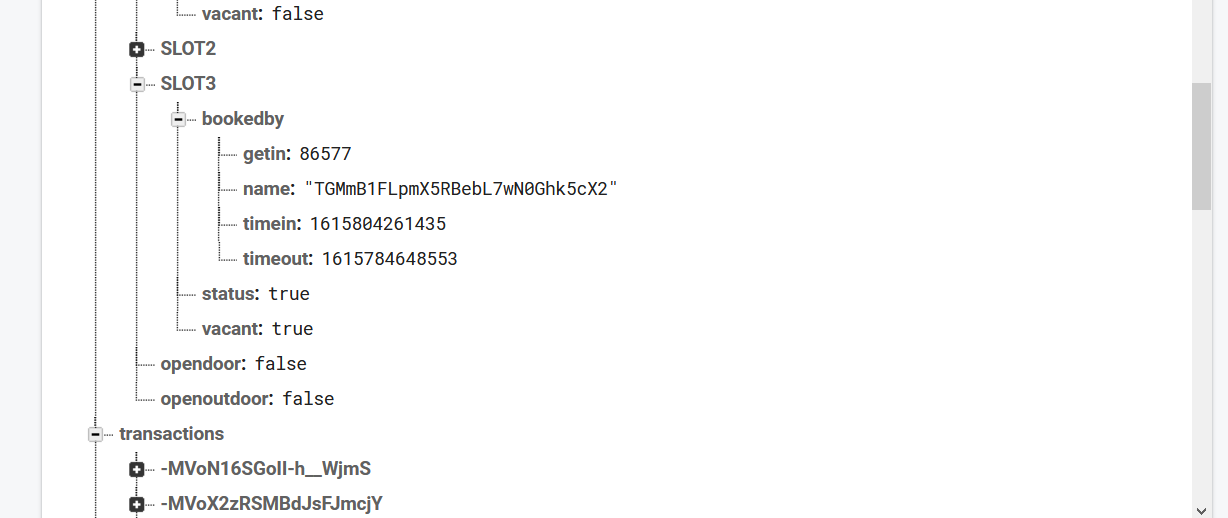


Firebase setup

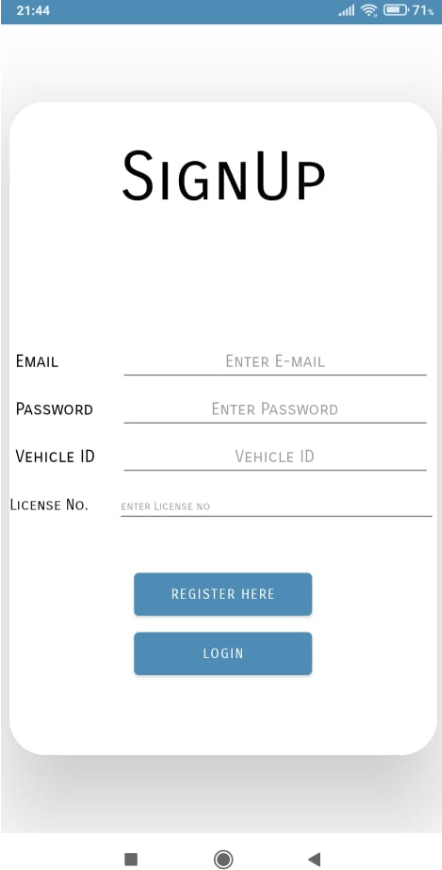


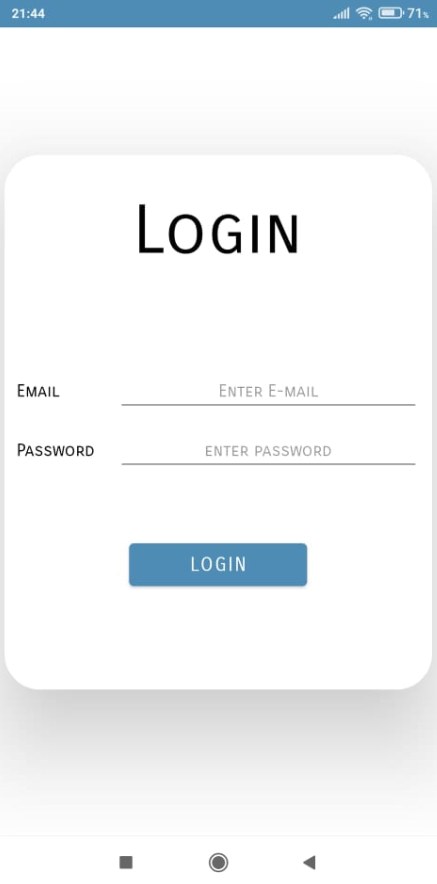


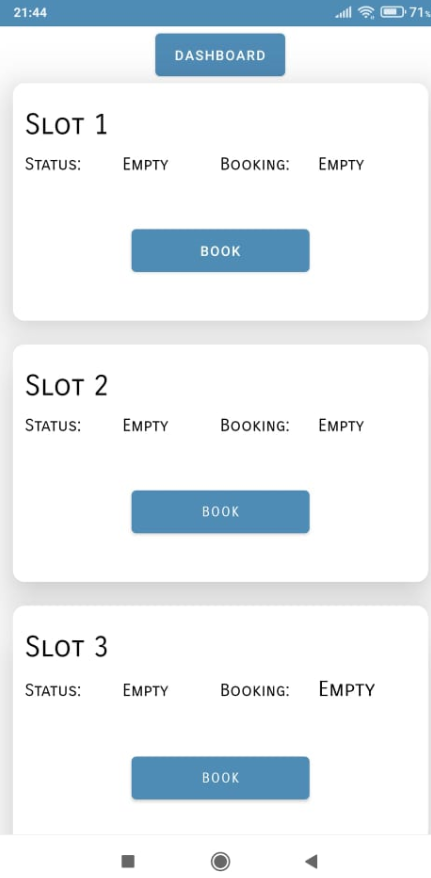


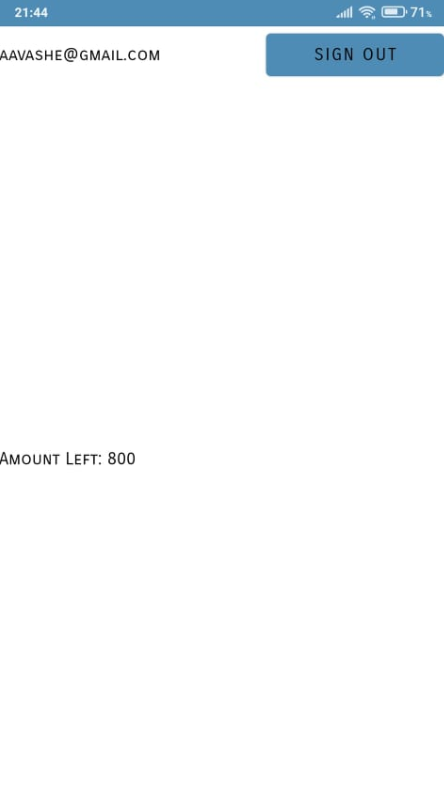


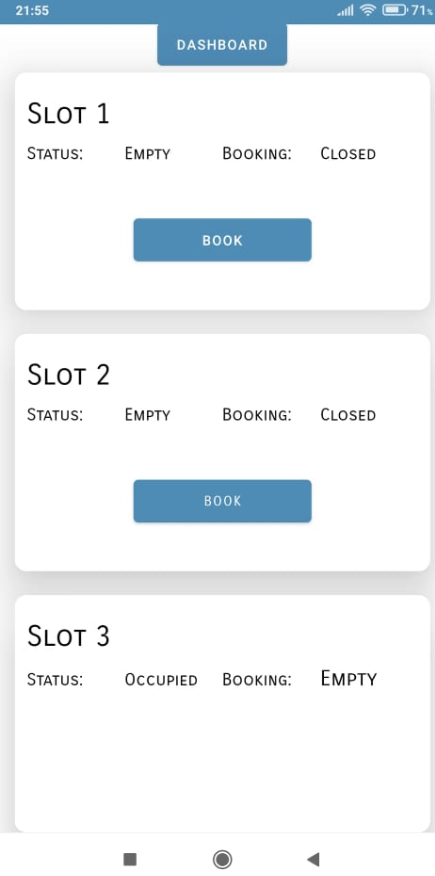
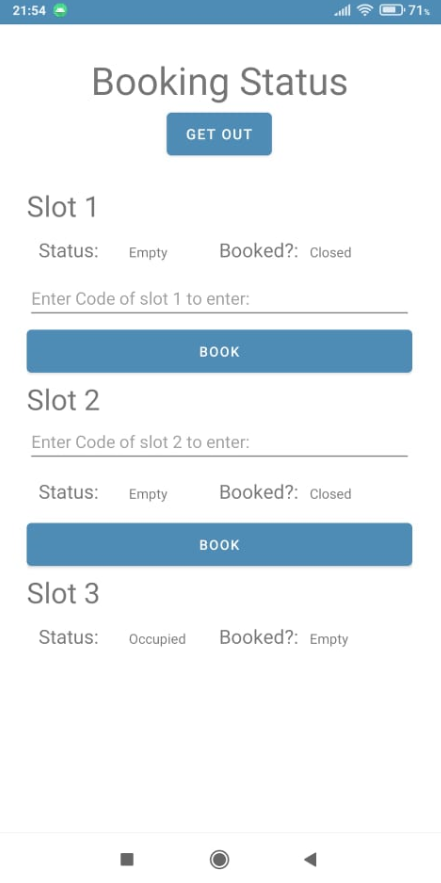
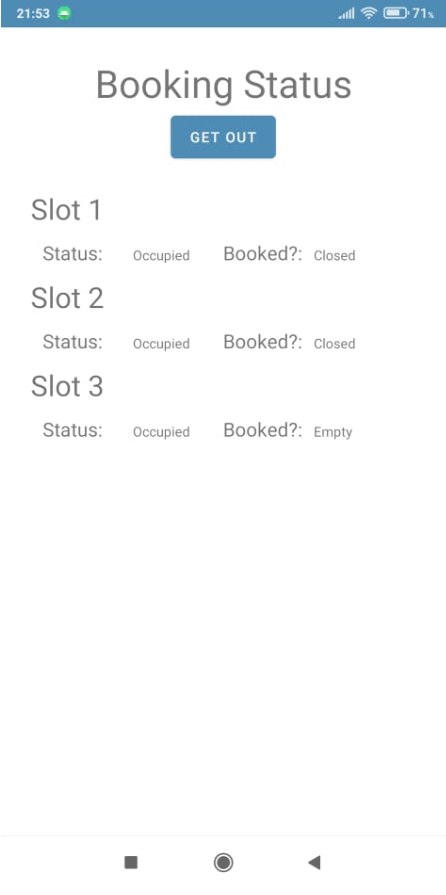
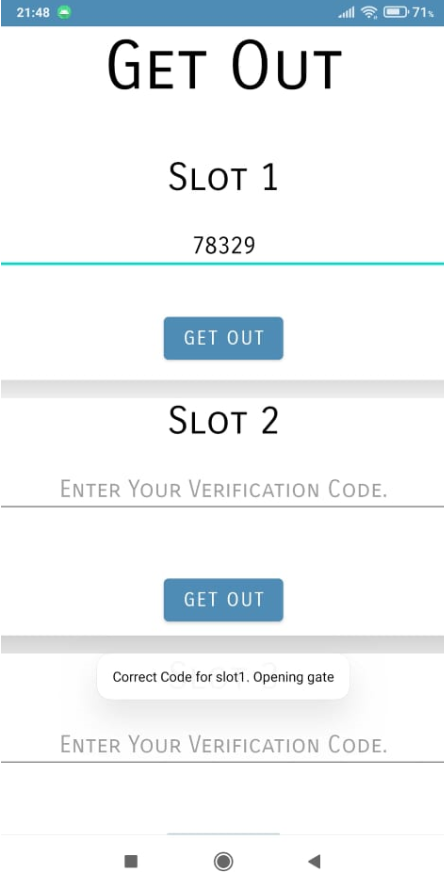
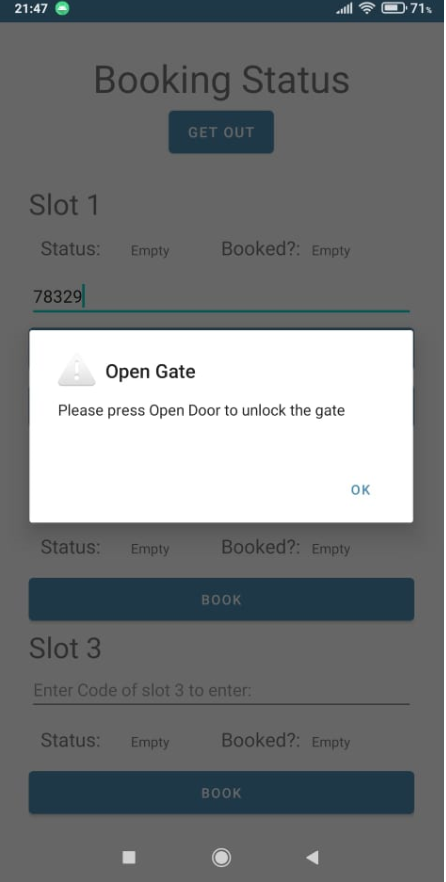
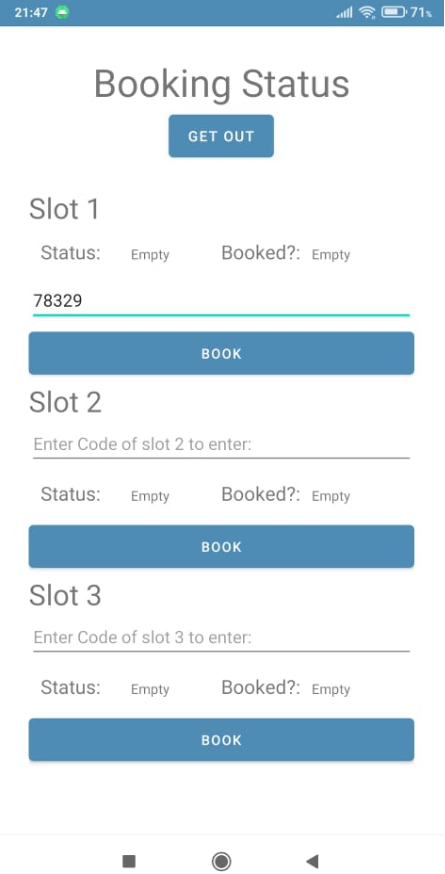
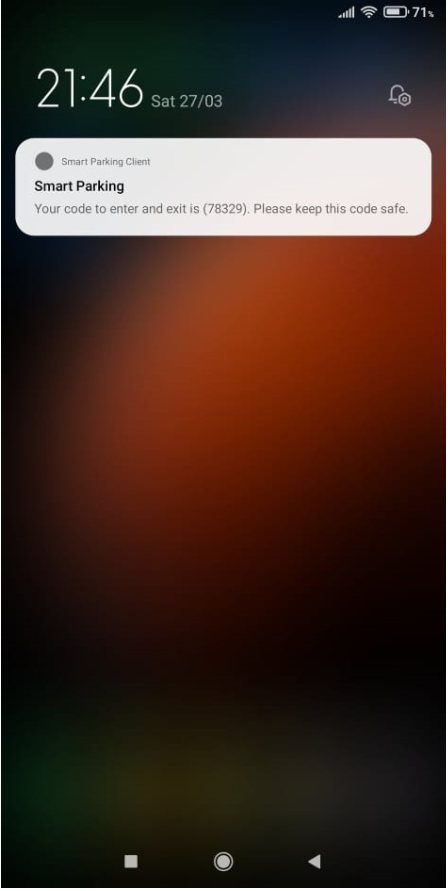
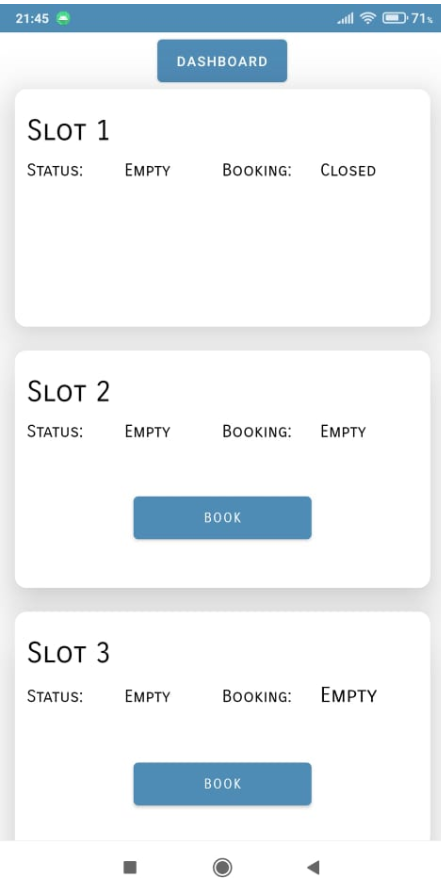
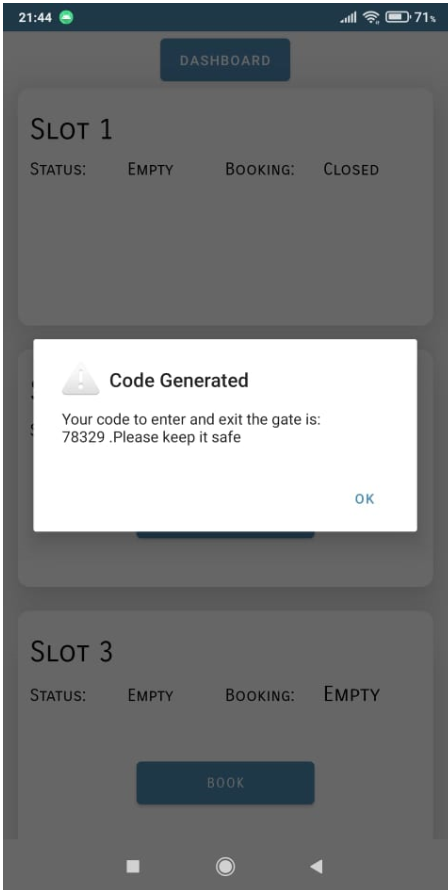
## Android App

****

****

****

****

****