SCHOOL OF INFORMATION TECHNOLOGY AND ENGINEERING

M.TECH(SOFTWARE ENGINEERING)

IOT BASED SMART PARKING LOT

Tharaka J
(18MIS0292)

Project Report
of
SWE4005-Internet of Things

Winter 2021-22

Submitted to
Faculty : Dr.Kavitha B.R

Signature:
Date:

TABLE OF CONTENTS

ABSTARCT

Parking problems are a common problem in most major cities. Our main aim of this project is to reduce the traffic in the parking place. Normally we can see in the multiplexes, cinema halls, large industries, and function halls there is problem they have to go and search which line is empty and which line having place to park the vehicle, for parking then they need workers for parking in correct position it is the money consumed process. So, to avoid this problem Car Parking System project is implemented. Our goal is to construct a car park system which is equipped with sensors and provides surveillance. We are developing a novel miniaturized modular platform for wireless sensor networks. The system architecture, hardware and software will be discussed as well as details of the deployment scenario chosen for the prototype of acar park management system. Some of the recent studies shows about the parking management and the slot management. And also gives the information about reservation based parking management. Chi-Hung Chuang, Luo-Wei Tsai, developed a monitoring system for parking lot management system and the result of access management is reduced human resource, through the recognition car license. The constraint of this project is the recognition process takes more time to compare. Mingkai Chen developed a parking guidance and information system based on wireless sensor system and the information is transmitted between the nodes and processing the data, and the information passes to the display drivers. In this the constraint is, if the main node of the sensor system fails means the total block is failed. Huang Cai-mei. Presented an idea for reserving the parking slots and reversed cars look for the intelligent terminals to achieve the parked position of vehicles and get the guide route, so that user can quickly find the parking area. BhosaleSwapnali developed an idea for generating the multiple images using a fixed camera capture under different variations. Multiple images detection & recognition is important in the analysis of video data and higher-level security system. Vanessa W.S. Tang presented an idea on WSN-based intelligent car parking system and the sensors are deployed into a car park field, with each parking lot equipped with one sensor node, which detects and monitors the occupation of the parking lot.

INTRODUCTION

The Main Objective is the Sensors were placed at the Entrance of the Popular areas orplaces like shopping mall, Cinemas, Schools, College, especially for Hospital, etc. It senses the car for parking and Checks whether the free slots were available for parkingif YES, the entry gate were automatically open and allows the Car to get inside, if NO,the entry gate remains close and display the message slots were full, so that the Car driver need not come out of the vehicle to see, he/she can automatically looks for theanother parking lot, he/she can easily identify that whether free slots are available ornot without removing seat belt too which make more easier. The main requirement isInfrared (IR) sensors which is placed at the entrance of the parking lot and Ultrasonic sensors to detect the vehicle.

LITERATURE SURVEY

S.No	Paper Title	Author, Year	Description
1.	Smart parking system happiest minds	Aditya Basu, 2018	The monitoring module in the system includes ultrasonic sensors/ ambient light sensor which identifies the free parking spaces and transmits the Information to control unitthrough ZigBee
2.	RFID based car parking system	Anusooya G, Christy Jackson J, SathyarajasekaranK and Kumar Kannan 2016	This system uses infrared transmitter-receiver pairs that remotely communicate the status of parking occupancy tothe raspberry pi and displaysthe vacant slots on the displayat the entrance of the parking

3.	Automated underground carparking system.	Karan Upendrabhai Vyas, Adarash Kumar, Vadher Dhaval Hareshkumar 2017	The chain and sprocketmechanism is used for driving the parking platform. This model will be further useful for different branches of engineering in order to develop different types of automations like PLC, Micro controller and computerization.
4.	Iot based smart parking system.	Abhirup Khanna Rishi Anand, 2016	Smart Parking system consistsof an on-sitedeployment of an IoT module that is used tomonitor and signalize the state of availability of each single parking space. A mobile application is also provided thatallows an end user to check the availability of parking space and book a parking slot accordingly.
5.	Smart Car Parking System Solution forthe Internet of Things in Smart Cities	Wael Alsafery, B. A. (2018)	This paper proposes a smart car parking system that will assist users to solve the issue of finding a parking space and to minimise the time spent in searching for the nearest available car park. In addition, it provides users with roads traffic congestion status.
6.	Iot based car parking management system with an application program interface.	Dr.Sathish kumar, Ravi Lakshmanan, Chandrasekharan Natraj 2019	The parking system is designed based on zone control architecture using NodeMCU, thus allowing easy management and implementation. An eight hexadecimal character is generated for both entry and exit access codes, and sent to the driver's email and mobile phone via SMS. Parking information are relayed and stored in the server using RESTful API implemented using Laravel PHP technology

7.	IoT Based Smart Parking System	Mohd Mustari Syafiq Ismail, 2019	The System is developed to approach the user to using application to review available parking space and pay parking through mobile phone application and developmonitoring system that detect presence vehicle using ultrasonic sensor and camera artificial intelligence
8.	A Prototype for IoT based Car Parking Management Systemfor Smart Cities.	M Kumar Gandhi and M. Kameswara Rao, 2019	This system helps user to find parking space availability withthe help of Internet of Things (IoT) technology by providing parking free space information. The IoT maintains the database of the parked vehicles through a shared server. So drivers can book the slots in advance and the parking information updated in serve.
9.	Smart Car Parking Management System Using IoT	Aniket Gupta, Sujata Kulkarni, 2020	A Smart Parking Management System that helps users to automatically find a free parking space with a smaller amount.Smart Parking involves use ofUltrasonic sensor, Arduino Uno,ESP8266-01 Wi-Fi Module, Cloud server. IOT based platform which enable to connect, analyze and automate data gathered from devices
10.	An approach to IoT based car parking and reservation system on Cloud.	Vaibhav Hans, Parminder Singh Sethi; Jatin Kinra - 2018.	This proposed solution makes the ancient parking system smarter by leveraging the power of IoT and embedding it with the latest innovation of electronic sensors & computers.
11.	Smart Parking System using IoT	ElakyaR,Juhi Seth, Pola Ashritha, R Namith_- , 2019	Using internet of thingstechnology, an effective andsmart solution to automate the management of the parking

			system that allocates an efficient parking space. The Internet of Things gives wirelessaccess to the system, allowing the user to keep track of the parking area's availability. With a growth in the number ofvehicles on the road in metropolitan areas, traffic congestion has become a serious issue. The goal of this paper is to find a solution to this problem.
12.	Android Application for Smart Parkingusing IoT	Ashna Viji Alex, Amina Abdul Rasheed, Shaun Thomas, Salmanil Farisi, Ansia S - 2021	The user can park their car without the need for human intervention. This technologyalso assists users in saving time and increasing their productivity. We propose an Android application for a smart parking system based on IoT technologies in this research. For payment, the systememploys RFID technology. Number plate recognition ensures the user's safety. Thegoal of this strategy is to reducethe time spent looking for a parking spot while simultaneously allowing for hands-free payment. This smart parking system can be used in parks, shopping malls, andhospitals. This application allows users to reserve parking spaces ahead of time.
13.	IoT Based Smart Parking System	Saidur Rahman, Poly Bhoumik. -2019	With the rapid growth of the population, traffic congestion is now at an all-time high. The useof personal vehicles has also increased in proportion to the population. Traffic congestion on the road has increased as a

			result of the increased use of automobiles. The majority of individuals prefer personal autos to public transportation. Finding a parking place in most metropolitan locations andcommercial sectors is extremely difficult and time-consuming, especially during rush hours. Finding a suitable and secure parking space is often costly in practically every major city across the world. The suggested concept is a smart parking system that provides information to users who are looking for a parking spot on the internet. It reduces the amount of time spent looking for a parking space in a parking lot.
14.	Smart Parking System using IOT	Waleed Zahir AlQaidhi, Muhammad Sohail 2020	The development of road infrastructure has resulted in a large growth in the number of private automobiles, resulting in traffic congestion, which has a direct impact on traffic flow and citizen life. In metropolitan places, parking has become aserious issue. Automatic car parking systems have been proposed to reduce the amountof space or size required for parking, particularly in congested areas with limited parking spaces, such as a multi- story car park that provides parking on multiple levels stacked vertically to increase the number of parking spaces.
15.	A Review on Smart IoT Based Parking System	Hafeez Sidd-2019	Various models enhanced with sensors, integrating cloud and mobile application are given in the investigations, resulting in a

			smart parking system that saves time, energy, gasoline, and carbon footprint. It has been observed that disabled persons are overlooked in their systems,and there is currently no unifiedglobal parking solution. The rapid advancement in the miniaturisation and robustness of processors, sensors, and machine learning coupled with the potential of a smart and single global solution for future researchers to address parking issues in both indoor andoutdoor parking lots has added to the potential of a smart and single global solution.

SYSTEM REQUIREMENTS

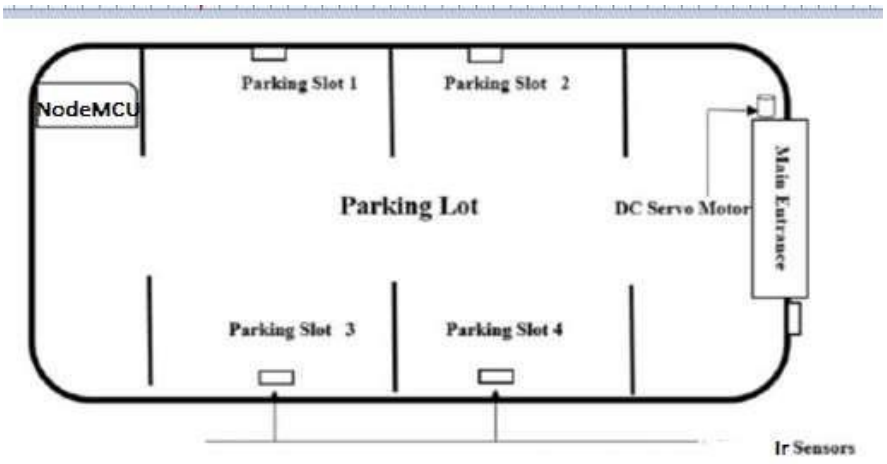
HARDWARE

- NodeMCU ESP8266
- IR Sensor
- Servo Motor

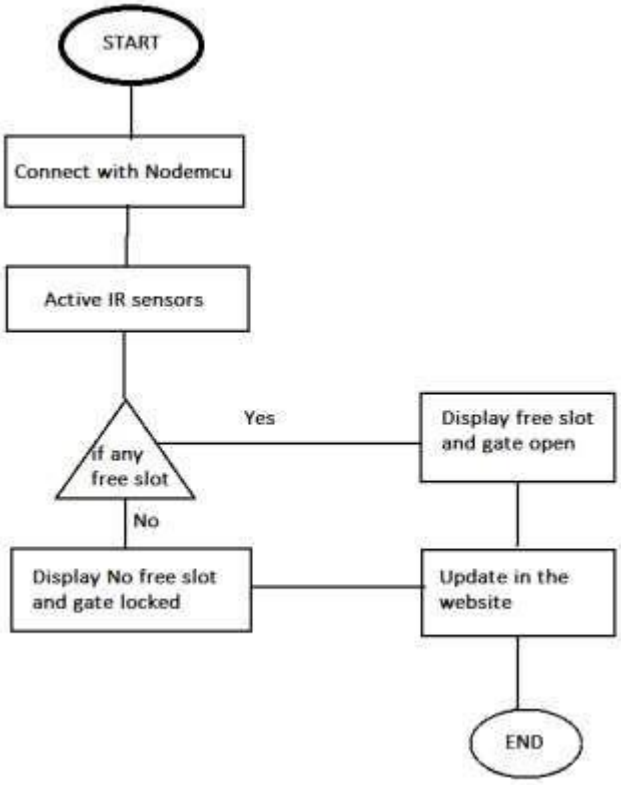
SOFTWARE

- Adafruit IO

DESIGN METHODOLGY:

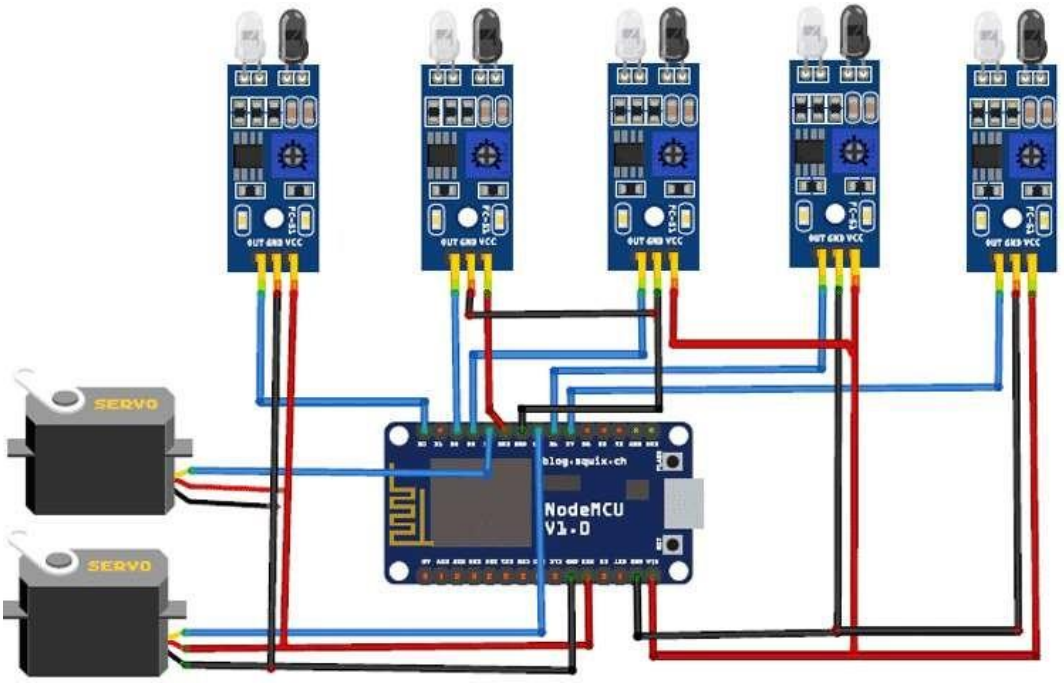


Flow chart:



IMPLEMENTATION:

Hardware Part



CODE:

First, include all the required libraries. **ESP8266 Wi-Fi** and **Servo.h** libraries are already installed in the IDE. You can download the NTP client and Adafruit MQTT libraries from the below links:

NTPClient.h

Adafruit_MQTT.h

```
#include <ESP8266WiFi.h>
#include <Servo.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <NTPClient.h>;
#include <WiFiUdp.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
```

Then include the Wi-Fi and Adafruit IO credentials that you copied from the Adafruit IO server. These will include the **MQTT server, Port No, User Name and AIO Key**

```
const char *ssid =  "WiFi Name";  // Enter your WiFi Name
const char *pass =  "Password"; // Enter your WiFi Password
String time1= "89";
#define MQTT_SERV "io.adafruit.com"
#define MQTT_PORT 1883
#define MQTT_NAME "User Name"
#define MQTT_PASS "AIO Key"
```

Set up the feed you're publishing to. Here *Agriculture Data* is the feed name.

```
Adafruit_MQTT_Subscribe EntryGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME "/f/EntryGate");
Adafruit_MQTT_Subscribe ExitGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME "/f/ExitGate");
……..
……..
```

Connect the Entry and Exit Servo Motor to the D4, D5 Pins of the NodeMCU, and select the out pins of IR sensor as INPUT.

```
myservo.attach(D4);
myservos.attach(D5);
pinMode(carExited, INPUT);
pinMode(carEnter, INPUT);
pinMode(slot1, INPUT);
pinMode(slot2, INPUT);
pinMode(slot3, INPUT);
```

Inside the *void loop*, ***timeClient.update()*** function is used to update the date and time whenever we request to NTP servers. After getting the data, we store the hour, minute and second in three different integers.

```
timeClient.update();
hh = timeClient.getHours();
mm = timeClient.getMinutes();
ss = timeClient.getSeconds();
```

Digitally read the entry and exit IR sensor pins and check if these pins are high. If pins are high, then move the servo motor to open the entry and exit gate. Then increase the count for entry gate and decrease the count for exit gate and publish the data to the Adafruit IO dashboard.

Check the slot 1 IR sensor. If it is ‘1’ and Boolean function is false, then get the entry time from the NTP server and save it in EntryTimeSlot1 variable. Publish the variable data to the Adafruit IO feed.

```
if (s1 == 1 && s1_occupied == false) {
        EntryTimeSlot1 =  h +" :" + m;
        s1_occupied = true;
        if (! EntrySlot1.publish ((char*) EntryTimeSlot1.c_str())){}
    }
```

If the IR sensor pin change to zero and Boolean function is true then publish the exit time to Adafruit IO feed.

```
if(s1 == 0 && s1_occupied == true) {
       ExitTimeSlot1 =  h +" :" + m;
       s1_occupied = false;
        if (! ExitSlot1.publish((char*) ExitTimeSlot1.c_str())){}
    }
```

Do similar steps as above for slot2 and slot 3 sensors.

```
if (s2 == 1&& s2_occupied == false) {
      EntryTimeSlot2 =  h +" :" + m;
      s2_occupied = true;
      if (! EntrySlot2.publish((char*) EntryTimeSlot2.c_str())){}
    }
  if(s2 == 0 && s2_occupied == true) {
       ExitTimeSlot2 =  h +" :" + m;
       s2_occupied = false;
        if (! ExitSlot2.publish((char*) ExitTimeSlot2.c_str())){}
  }
  if (s3 == 1&& s3_occupied == false) {
      EntryTimeSlot3 =  h +" :" + m;
      s3_occupied = true;
       if (! EntrySlot3.publish((char*) EntryTimeSlot3.c_str())){}
    }
```

```
if(s3 == 0 && s3_occupied == true) {
      ExitTimeSlot3 =  h +" :" + m;


     s3_occupied = false;
      if (! ExitSlot3.publish((char*) ExitTimeSlot3.c_str())){ }
 }
```

Here we are directly checking for a specific word in our subscribed feed, and if the word matches with our specified word, i.e., 'ON,' it will rotate the servo motor to open the gate.

```
if (subscription == &EntryGate)
    {
     Serial.println((char*) EntryGate.lastread);
  if (!strcmp((char*) EntryGate.lastread, "ON"))
     {
      myservos.write(OPEN_ANGLE);
      delay(3000);
      myservos.write(CLOSE_ANGLE);
   }
```

Do similar steps as above for the Exit gate**.**

This is how the parking details are published on the Adafruit IO dashboard. It will show the entry time and exit time for every slot. This dashboard also has two buttons to manually open the entry and exit gate.

So this is how a Smart Parking System using IoT can be built. You can add more sensors to increase the parking slots and can also add a payment system to automatically pay the parking fee. Comment below if you have any doubts regarding this project.

```
#include <ESP8266WiFi.h>
#include <Servo.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <NTPClient.h>;
#include <WiFiUdp.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
const char *ssid =  "Galaxy-M20";      // Enter your WiFi Name
const char *pass =  "ac312124"; // Enter your WiFi Password

#define MQTT_SERV "io.adafruit.com"
#define MQTT_PORT 1883
#define MQTT_NAME "aschoudhary"
```

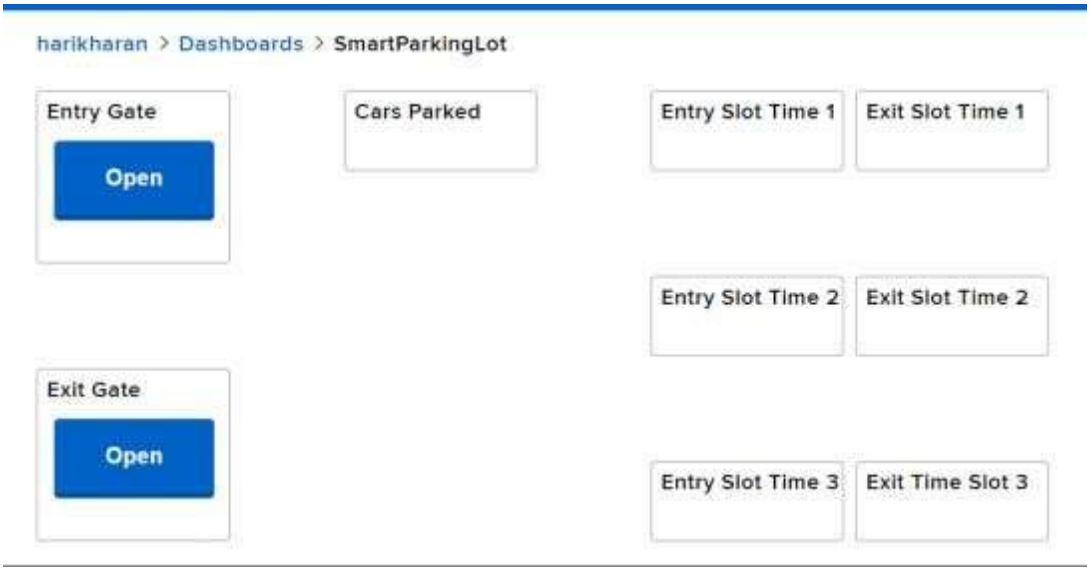
```
#define MQTT_PASS "1ac95cb8580b4271bbb6d9f75d0668f1"
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 19800,60000);
Servo myservo;                    //servo as gate
Servo myservos;                         //servo as gate

int carEnter = D0;                 // entry sensor
int carExited = D2;               //exi sensor
int slot3 = D7;
int  slot2  =  D6;
int slot1  = D3;
int count =0;
int CLOSE_ANGLE = 80;  // The closing angle of the servo motor arm
int OPEN_ANGLE = 0;  // The opening angle of the servo motor arm
int  hh, mm, ss;
int pos;
int pos1;

String h, m,EntryTimeSlot1,ExitTimeSlot1, EntryTimeSlot2,ExitTimeSlot2, EntryTimeSlot3,ExitTimeSlot3;
boolean entrysensor, exitsensor,s1,s2,s3;

boolean s1_occupied = false;
boolean s2_occupied = false;
boolean s3_occupied = false;

WiFiClient client;


Adafruit_MQTT_Client mqtt(&client, MQTT_SERV, MQTT_PORT, MQTT_NAME, MQTT_PASS);

//Set up the feed you're subscribing to
Adafruit_MQTT_Subscribe EntryGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME "/f/EntryGate");
Adafruit_MQTT_Subscribe ExitGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME "/f/ExitGate");

//Set up the feed you're publishing to
Adafruit_MQTT_Publish CarsParked = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/CarsParked");
Adafruit_MQTT_Publish EntrySlot1 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/EntrySlot1");
Adafruit_MQTT_Publish ExitSlot1 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/ExitSlot1");
Adafruit_MQTT_Publish EntrySlot2 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/EntrySlot2");
Adafruit_MQTT_Publish ExitSlot2 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/ExitSlot2");
Adafruit_MQTT_Publish EntrySlot3 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/EntrySlot3");
Adafruit_MQTT_Publish ExitSlot3 = Adafruit_MQTT_Publish(&mqtt,MQTT_NAME "/f/ExitSlot3");

void setup() {
  delay(1000);
  Serial.begin (9600);
  mqtt.subscribe(&EntryGate);
  mqtt.subscribe(&ExitGate);
  timeClient.begin();
  myservo.attach(D4);       // servo pin to D6
  myservos.attach(D5);        // servo pin to D5
  pinMode(carExited, INPUT);    // ir as input
  pinMode(carEnter, INPUT);     // ir as input
  pinMode(slot1, INPUT);
  pinMode(slot2, INPUT);
  pinMode(slot3, INPUT);
  WiFi.begin(ssid, pass);                              //try to connect with wifi
  Serial.print("Connecting to ");
  Serial.print(ssid);                         // display ssid
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print(".");                        // if not connected print this
    delay(500);
  }
  Serial.println();
```

```
 Serial.print("Connected to ");
 Serial.println(ssid);
 Serial.print("IP Address is : ");
 Serial.println(WiFi.localIP());                    //print local IP address
}

void loop() {

 MQTT_connect();
 timeClient.update();
 hh = timeClient.getHours();
 mm = timeClient.getMinutes();
 ss = timeClient.getSeconds();
 h= String(hh);
 m= String(mm);
 h +" :" + m;

 entrysensor=  !digitalRead(carEnter);
 exitsensor = !digitalRead(carExited);
 s1 = digitalRead(slot1);
 s2 = digitalRead(slot2);


 s3 = digitalRead(slot3);



  if (entrysensor == 1) {                 // if high then count and send data
  count=  count+1;                        //increment count
  myservos.write(OPEN_ANGLE);
  delay(3000);
  myservos.write(CLOSE_ANGLE);
  }

  if (exitsensor == 1) {                    //if high then count and send
  count= count-1;                           //decrement count
  myservo.write(OPEN_ANGLE);
  delay(3000);
  myservo.write(CLOSE_ANGLE);
 }
 if (! CarsParked.publish(count)) {}

 if (s1 == 1 && s1_occupied == false) {
       Serial.println("Occupied1 ");
       EntryTimeSlot1 =  h +" :" + m;
       //Serial.print("EntryTimeSlot1");
       //Serial.print(EntryTimeSlot1);

       s1_occupied = true;
       if (! EntrySlot1.publish((char*) EntryTimeSlot1.c_str())){}
   }
 if(s1 == 0 && s1_occupied == true) {
      Serial.println("Available1 ");
      ExitTimeSlot1 =  h +" :" + m;
      //Serial.print("ExitTimeSlot1");
      //Serial.print(ExitTimeSlot1);

      s1_occupied = false;
       if (! ExitSlot1.publish((char*) ExitTimeSlot1.c_str())){}
}
 if (s2 == 1&& s2_occupied == false) {
     Serial.println("Occupied2 ");
     EntryTimeSlot2 =  h +" :" + m;
```

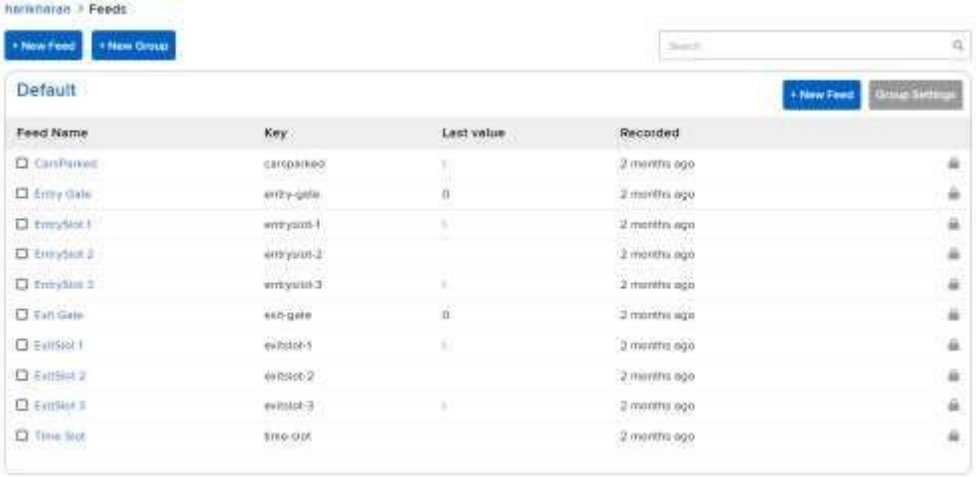
```
    //Serial.print("EntryTimeSlot2");
    //Serial.print(EntryTimeSlot2);

    s2_occupied = true;
    if (! EntrySlot2.publish((char*) EntryTimeSlot2.c_str())){}
  }
 if(s2 == 0 && s2_occupied == true) {
     Serial.println("Available2 ");
     ExitTimeSlot2 =  h +" :" + m;
     //Serial.print("ExitTimeSlot2");
     //Serial.print(ExitTimeSlot2);

     s2_occupied = false;
      if (! ExitSlot2.publish((char*) ExitTimeSlot2.c_str())){}


 }
 if (s3 == 1&& s3_occupied == false) {
    Serial.println("Occupied3 ");
    EntryTimeSlot3 =  h +" :" + m;
   //Serial.print("EntryTimeSlot3:  ");
    //Serial.print(EntryTimeSlot3);
    s3_occupied = true;
     if (! EntrySlot3.publish((char*) EntryTimeSlot3.c_str())){}
  }
 if(s3 == 0 && s3_occupied == true) {
     Serial.println("Available3 ");
     ExitTimeSlot3 =  h +" :" + m;
     //Serial.print("ExitTimeSlot3:  ");
     //Serial.print(ExitTimeSlot3);
     s3_occupied = false;
      if (! ExitSlot3.publish((char*) ExitTimeSlot3.c_str())){ }
 }

 Adafruit_MQTT_Subscribe * subscription;
 while ((subscription = mqtt.readSubscription(5000)))
   {

  if (subscription == &EntryGate)
    {
     //Print the new value to the serial monitor
     Serial.println((char*)  EntryGate.lastread);

  if (!strcmp((char*) EntryGate.lastread, "ON"))
     {
      myservos.write(OPEN_ANGLE);
      delay(3000);
      myservos.write(CLOSE_ANGLE);
    }
}
  if (subscription == &ExitGate)
    {
     //Print the new value to the serial monitor
     Serial.println((char*)  EntryGate.lastread);

  if (!strcmp((char*) ExitGate.lastread, "ON"))
     {
      myservo.write(OPEN_ANGLE);
      delay(3000);
      myservo.write(CLOSE_ANGLE);
    }
}
```

```
}
}
void MQTT_connect()
{
  int8_t ret;

  // Stop if already connected.
  if (mqtt.connected())
  {
    return; }  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0) // connect will return 0 for connected
  {
      mqtt.disconnect();
      delay(5000);  // wait 5 seconds
      retries--;
      if (retries == 0)
      {
        // basically die and wait for WDT to reset me
        while (1);
```

IMPLEMENTATION IN WEB SERVER:



SCREENSHOTS:



RESULT:

The demand of smart parking system is increasing significantly. This allows user to involve real time access of the availability of the parking space. The existing system in today's world doesn't contains the facilities of parking reservation and parking slot availability checker. The existing system was vision-based monitoring system [7] which estimates the number of the parking slots available in the area by counting the number of incoming and outing cars which consumes lot of time and efforts. The next existing system was sensor-based system which uses ultrasonic sound waves for detecting the presence of vehicles and then two-tier parking came into existence which used the concept of parking cars one above another. The result of the paper is to make the parking area connected with the world as well as reduces time and can be cost effective for the user. The result of this paper is to reduce car theft. This paper reduces overall fuel energy of the vehicle which is consumed in the search of the car.

CONCLUSION:

The growth of Internet of Things have given rise to New possibilities in terms of smart cities. Smart parking facilities and traffic management systems have always been at the core of constructing smart cities. In this paper, we address the issue of parking and present an IoT based web application smart parking system. The system that we propose provides real time information regarding availability of parking slots in a parking area. Users from any locations could book a parking slot for them by the use of our web application. The efforts made in this paper are indented to improve the parking facilities of a city and thereby aiming to enhance the quality of life of its people. In our system user can view the real view of parking slot of nay register buildings, mall, hospitals, colleges and may more public parking areas. Due to this user has a choice that in what transport system he should use to visit that place.

REFERENCES:

1) https://www.researchgate.net/publication/333575897_Iot_based_car_parking_management_system_with_an_application_program_interface
2) https://ieeexplore.ieee.org/abstract/document/7380487
3) https://www.sciencepublishinggroup.com/journal/paperinfo?journalid=325&doi=10.11648/j.ajset.20170204.13
4) https://indjst.org/articles/a-prototype-for-iot-based-car-parking-management-system-for-smart-cities
5) https://iopscience.iop.org/article/10.1088/1742-6596/1424/1/012021/pdf
6) https://www.happiestminds.com/Insights/smart-parking/
7) https://www.cs.le.ac.uk/people/srm13/publications/iccais-2018-parking.pdf
8) https://research.vit.ac.in/publication/rfid-based-smart-car-parking-system
9) https://www.researchgate.net/publication/341870728_Smart_Parking_System_based_on_IOT
10) https://ieeexplore.ieee.org/document/7562735
11) https://www.ijeat.org/wp-content/uploads/papers/v9i1/A1963109119.pdf

12) https://www.ijert.org/android-application-for-smart-parking-using-iot

13) https://www.researchgate.net/publication/329686583_IoT_Based_Smart_Parking_System
14) https://www.jsr.org/index.php/path/article/view/881
15) https://www.researchgate.net/publication/337819752_A_Review_on_Smart_IoT_Based_Parking_System