Name: Ponsivalingraj Roll no: 950421104036

Subject topics: Internet Of Things Date: 01/11/2023

Phase 5: Project Documentation & Submission

.SMART PARKING SYSTEM.

Introduction:

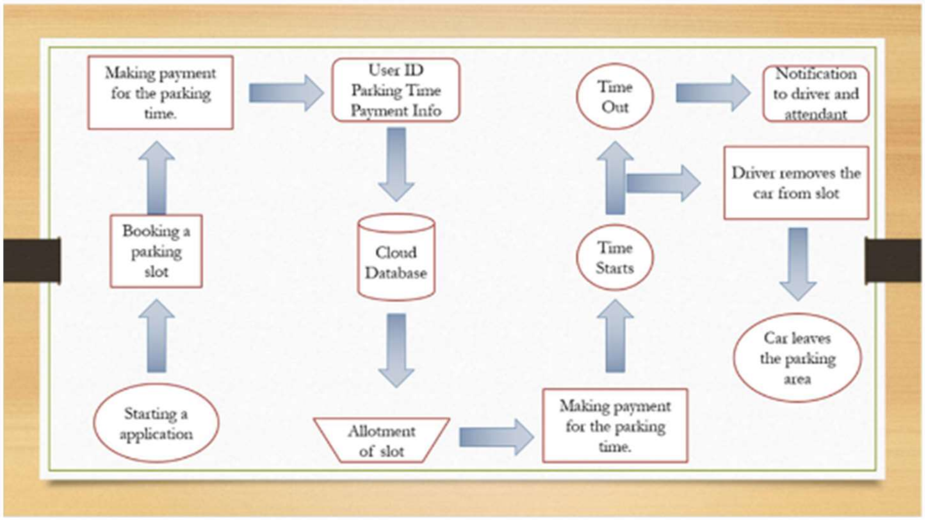
The idea of creating a Smart City is now becoming possible with the emergence of the Internet of Things. One of the key issues that smart cities relate to are car parking facilities and traffic management systems[1]. In a recent research it has been found thatfinding an available parking spot is always difficult for drivers. Ittends to become harder with the increasing number of private car users. This concern attracted strategic investments from dedicated industry sectors to boost parking revenues through technology-enabled solutions like reduction in searching times, traffic congestion and road accidents. Problems related to parking and traffic congestion can be solved if the drivers can be informed in advance about the availability of parking spaces before reaching to their intended destination. Researchers [2] show that more than 66% of drivers are willing to pay for car parking during working hours. This directly adds value to the car parking business. It is stimulus for the development of intelligent car parkingservices for smart cities. The system helps a user know the availability of parking spaces. The parking systems aims at providing facilities to users like find, allocate, and reserve the „best" available car parking lot for a user in a particular area, providing navigation instructions for reaching this lot.

Problem Definition and Design Thinking

What is Smart Parking?

Smart parking system using IoT has smartphones and other sensors added into an interconnected system to determine parking space or level and provide real-time feedback. It is accomplished by using cameras, counters on the doors or gates of parking lots, sensors embedded in the paved area of individual parking lots, among other locations, depending on the deployment. Solutions using IoT-based parking

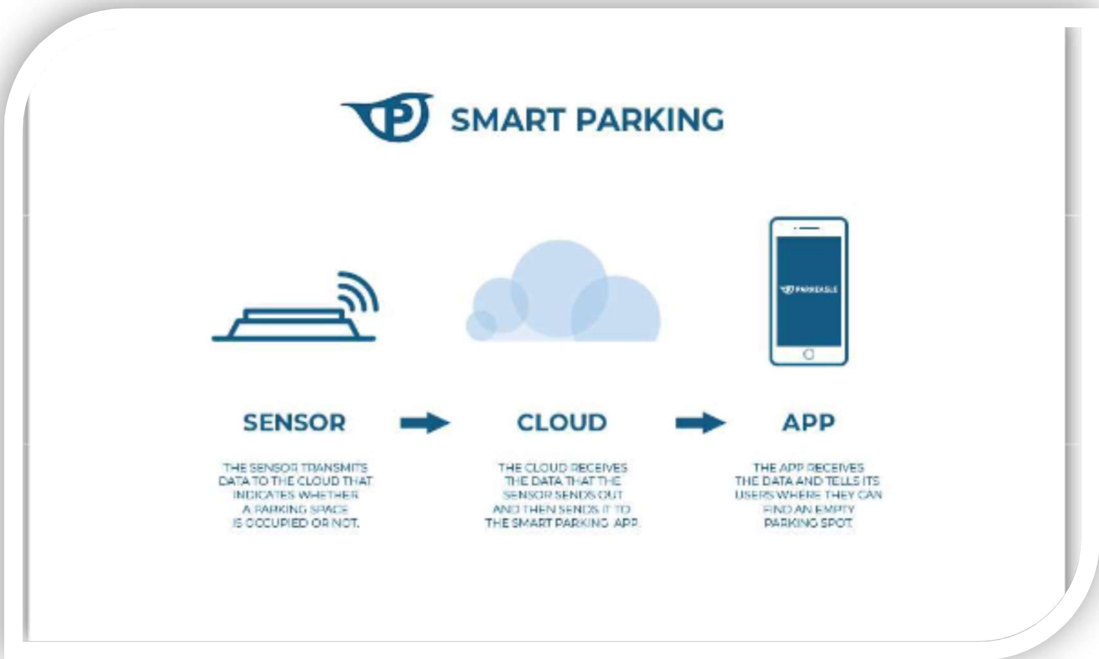
IoT-based smart parking system transmits available and occupied parking spaces via a web/mobile application.
Each parking space has an IoT gadget, which includes sensors and microcontrollers. The user gets real-time updates on the availability of all parking spaces and, therefore, an option to choose the best one. This solution alone initiates a chain-reaction of benefits, from lesser traffic congestion to reduced fuel efficiency, in urban areas where parking is often painstaking.



Smart Parking is a parking solution that can include in-ground Smart Parking sensors, cameras or counting sensors. These devices are usually embedded into parking spots or positioned next to them to detect whether parking bays are free or occupied. This happens through real-time data collection. The data is then transmitted to a smart parking mobile application or website, which communicates the availability to its users. Some companies also offer other in-app information, such as parking prices and locations. This gives you the possibility to explore every parking option available to you.

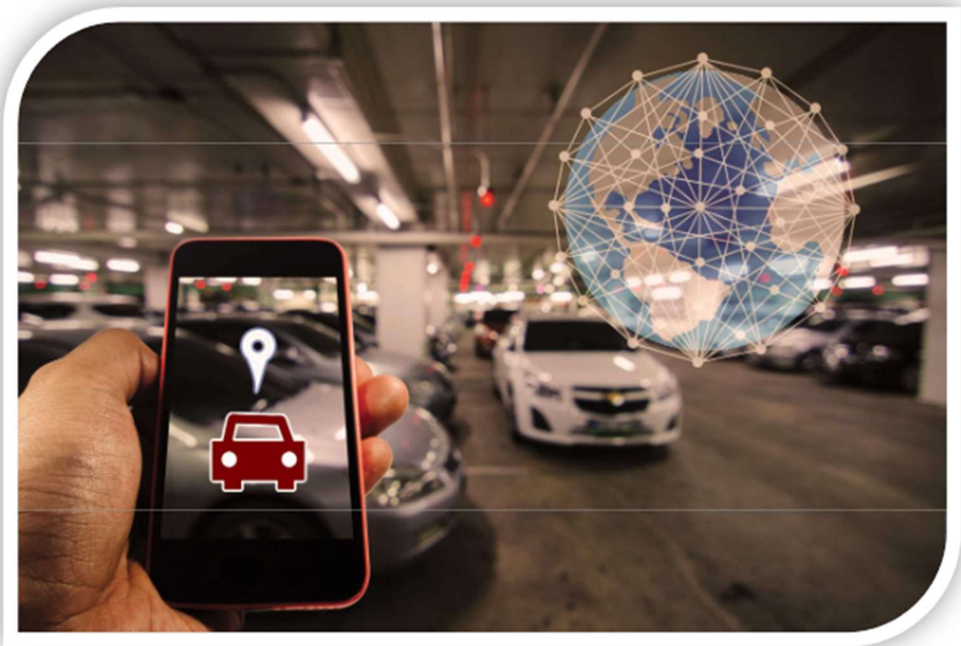
Smart Parking and its Smart Parking Sensors can be seen as a part of smart cities. These smart cities are cities that are driven by an IT infrastructure and by using this infrastructure, cities can enhance the quality of life and improve economic development for its inhabitants. Becoming a smart city can be a good way to collect historical data in a relatively easy way. By collecting this data, cities can analyze how processes, like parking can be optimized.

As a result of using Smart Parking, people who are looking to find a parking spot will find it in the most efficient way possible and companies or municipalities can optimize their parking territories. It also makes cities more livable, safer and less congested.



Advantages for Cities

- **Less pollution:** Smart Parking contributes to a cleaner environment. Reducing the time that is necessary to find a parking spot will reduce the amount of fuel that is used when looking for a parking space. This makes the process of finding a parking spot contribute to less pollution, which is beneficial for everyone.
- **The space of a municipality will be utilized more efficiently:** because Smart Parking sensors transmit live-data, drivers will have a real-time overview of the occupancy of parking bays. This means that free spots can be filled quicker, which will reduce the time that a parking spot is empty.
- **Safety:** The use of Smart Parking Sensors can optimize safety within cities. As a result of placing, for instance, on-ground sensors on parking bays, people will not be as stressed as when they are looking for parking spaces. Because these people will know where they are going, they can simply navigate to their parking spot and they will not have to stress out about it.
- **Real-time parking analytics for cities:** Parking space will become intelligent by use of the smart parking sensors on the parking bays. This means that as a city you're able to see historical data which is stored and you're able to make data driven decision and predictions based on the parking sensor data.



Applications of Smart Parking Systems using IoT

Smart cities offer better use of space, less traffic, clean air, and more efficient public services, increasing the quality of life. In addition, smart cities provide many jobs and economic opportunities, and strong social connections.
Smart parking IoT project will help in:

- **The seamless flowing of traffic**

  Public transport routes can be adjusted in real-time according to need, and smart traffic lights systems can improve congestion. cse smartparking

- **Energy efficiency can be improved**

  One can easily track down the power consumption & energy consumption by monitoring in real-time.

- **Cities can be made safer**

  Cities can use technology to improve residents' safety and improve response times with the widespread use of Wi-Fi communications and IoT technology.

- **Encouragement of greater citizen engagement**

  Citizens can respond to daily problems enabling neighbors to connect and share resources to improve communities and neighborhoods.

Innovation(smart parking)

Technology is the digital age. Since the invention of the microchip in 1958 technology has been on the rise to facilitate our way of life and living. The foundation of this innovation is *data*.
Data is our digital footprint, our breadcrumb trail. To make advancements in our digital innovations–to improve our cities and lifestyles, data is:

- Collected
- Mined
- Analyzed
- Applied.

1. Smart parking sensors

The idea is simple: install sensors throughout the city that communicate with an app to help drivers find free parking spots.
The realization, not so much: the ground needs to be dug up to install these sensors. **Inner city construction is notorious for creating traffic backlogs.** The

project will also *reduce* the number of available parking spaces during construction.

It's a project where the visionaries need to think in the long run. **Focus on the outcome rather than the process.** The local council will need to create a campaign to notify the public about the projected plan and objective.

Major cities such as San Francisco, Singapore, and Berlin already employ this technology - the pioneers of smart parking innovations.

2. Parking applications

Free parking apps are a *dream come true* for all car owners. The major apps available offer:

- Space availability
- Space reservation
- Reservation in advance
- Flexible reservation times
- Payment options - in advance
- Top-up payment options
- On-street, public, and private parking spaces
- Times-up reminders

3. Stacked parking

Double-decker parking solutions do just that–double your parking space. Their technology uses a simple elevation function. When you find your spot, you drive onto a ramp. When you exit your car you can flip a lever for the ramp to raise. **This frees up a parking space underneath.**

The downside to this is a higher level of *organization*. For employers, it's best to allocate parking spaces so those who leave later park first. This allows a smooth transition for car owners who leave earlier.

These options work best for:

  - Car dealerships
  - Mechanic workshops
  - Ferry crossings
  - Private residences

- Robot valet parking (more on this below)

The conception is indeed smart, but the realization can be tricky. The onus is to first *identify* your need, *evaluate* the current parking timings, and *proceed* to choose the best solution.

4. Subterra parking lifts

This smart parking solution is designed with homeowners in mind. After you park your car in your garage and exit it, **your car is lowered to a subterranean space**. Many private homes come with basements and underground garages. Store your cars here, choose which one you prefer to drive, mount the lift and drive away without a congested driveway.

Also popular with inner city parking garages. Employing *automated parking valets*, they can now utilize underground space without needing to spend too much on human navigation.

Your car will be automatically transported to an available parking spot underground, possibly tiered, optimizing space to the max, but more on that below.

5. Reduced fees for EVs

This innovation for smart parking promotes the use of electric vehicles. An objective for smart cities is to reduce carbon emissions. Therefore, it makes sense to offer lower parking rates for EVs. Smart move.

In conjunction with smart parking apps and designated EV charging stations, you can brave inner city parking with your EV.

If you don't need to charge your car each time you park it, you can apply for a residential parking permit at a *reduced cost*. There's no one price however, it fluctuates depending on local councils, but definitely worth looking into.

6. Robot valet parking

According to IEEE, Robotic Valet Automated Parking Systems (APS) is a viable solution to the parking problem. Designed initially for dedicated parking lots, **the automated parking service saves car owners time and gas looking for available space**.

When you enter the lot you drive your car onto a sort of loading dock. With the help of an app, you can leave your car and let the robotics do the rest.

The dock automatically transports your car to the nearest free space. This reduces accidents, saves space, can use parking lifts, and boosts the overall efficiency of the parking garage.

In the unlikely event of any mishaps, the services of a dedicated car accident lawyer are readily available to provide expert guidance and support.

7. Solar panels parking shades

Marrying smart parking and renewable energy, this innovation is just the tip of the iceberg. For outdoor parking lots, installing solar panels as shades offer the following:

- Parking safe from drastic weather conditions
- Reduced opportunity for bird dropping to dirty your car
- Production of energy while optimizing space

Lot owners can feed the energy back into the grid and benefit from local tax cuts. Or, if the parking area is operating an APS they break even with energy consumption.

These schools in California benefitted from reduced electricity fees for up to 20 years. Smart parking innovation at its finest! And it instills a sense of green forward-thinking in the students too to boot.

Smart Parking Lot Using Raspberry Pi

Hardware

- 2x Ultrasonic sensors
- 1x servo motor
- 4x white LED's
- 1x Light dependent resistor
- 1x 2*16 LCD Display

Computing

- 1x Raspberry Pi 3 B+
- 1x SD card (8-16GB the choice is yours)
- 1x Adafruit I2S 3W Class D Amplifier Breakout - MAX98357A
- 1x MCP3008
- 1x Resistor 2.2K Ohm
- 9x Resistors 1K Ohm

Various

- Jumper wires
- Breadboard
- UTP-cable
- Adapter for your raspberry pi

Software

- PuTTY
- Software to edit code(you can use whatever you want as long as it supports: Python, HTML, CSS and java script
- Win32 Disk Imager
- Raspbian OS image
- VNC Viewer (optional)

Step 1: Setup RPI

Before we get started with wiring the hardware, writing the

software and make the casing we will need to set up your PI with the necessary software you will need to make the project run.

You will need an image of Raspbian to put on your SD card and to write it to the SD card Win32 Disk Imager. You can find links to both down below.

- Win32 Disk Imager: https://sourceforge.net/projects/win32diskimager/

- Raspbian image: https://www.raspberrypi.org/downloads/raspbian/

The installation

1. Open Win32 Disk Imager
    1. Select the image you just downloaded via the folder icon
    2. Select your SD card via the drop down
    3. Click on write
    4. When the process is finished your computer will probably ask you if you want to format the SD card do not do this

When you are finished with these steps you will need to do some extra stuff so you can access the software of your PI.

1. Open the SD card in your explorer
2. Open the file "cmdline.txt"
3. You will see a long line of text at the end add: ip=169.254.10.1
4. Save the file
5. Create a file named ssh with no extension (if you don't know how to do this just click on new file, select new text file delete the extension and name it ssh)
6. The last step of this part is to safely eject the SD card and put it into the Raspberry PI

Step 2: Database

MariaDB

Run the following command to install MariaDB this is the database system required to keep track of all the data the sensors read in.

sudo apt install mariadb-server

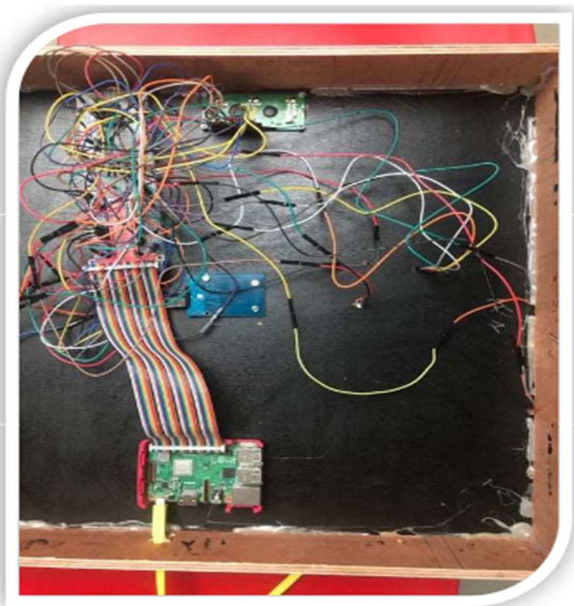
Now we will need to secure our installation

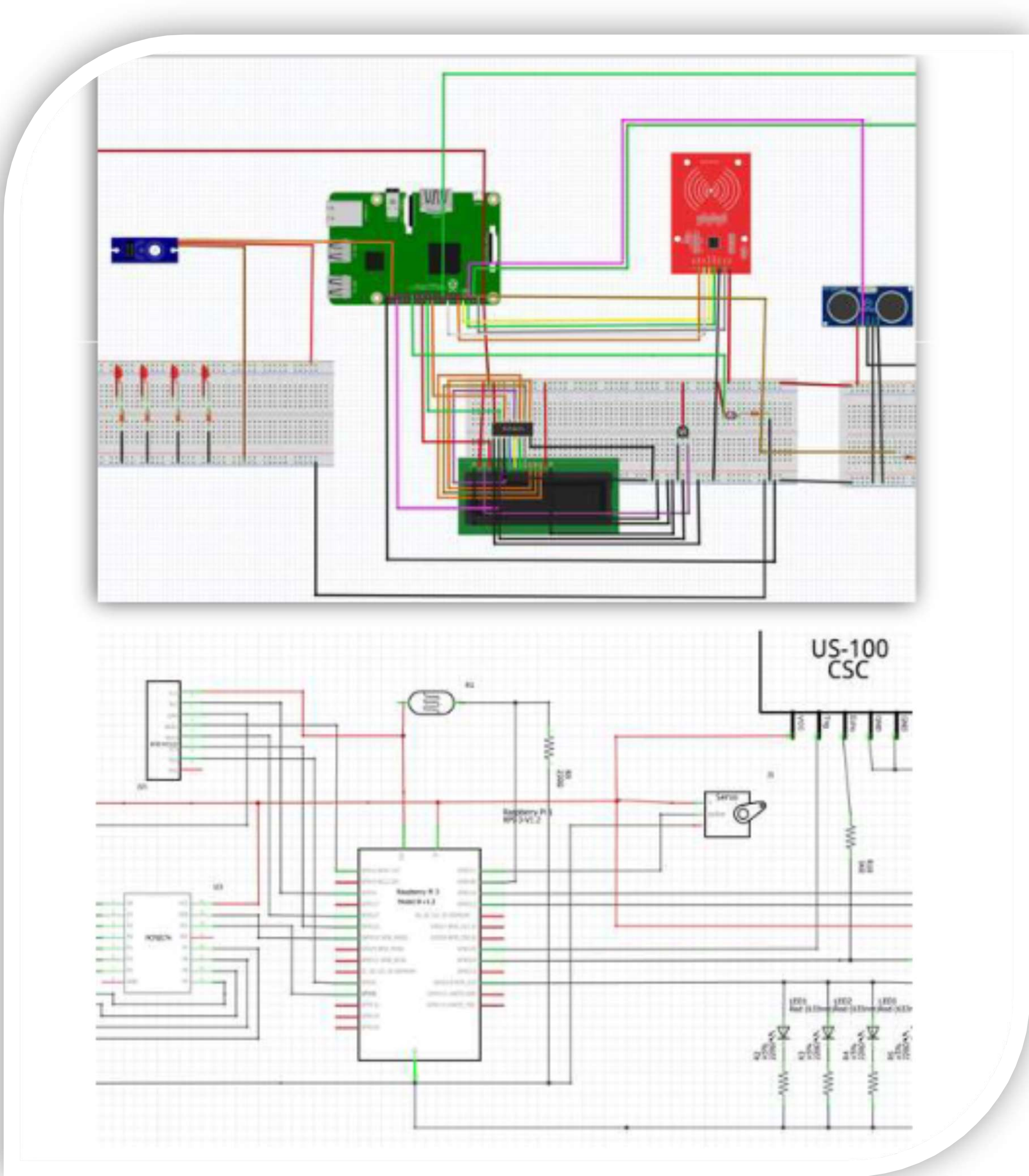
Mysql_secure_installation

Once you ran the command it will ask us for a root password we don't have one yet so just press enter.

Now it will ask you if you want one for security reasons we want one so press Y on your keyboard and choose whatever password you want.

Step 3: Wiring





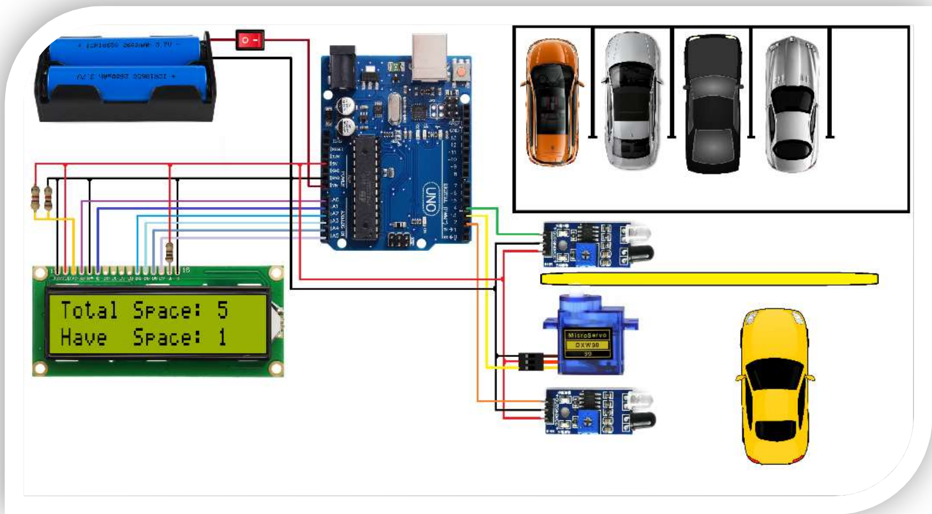
Hardware Required

- Arduino Uno
- 1602 LCD Display (optional)
- Power supply
- IR sensor

- [Micro Servo motor]()
- [Jumper cables]()

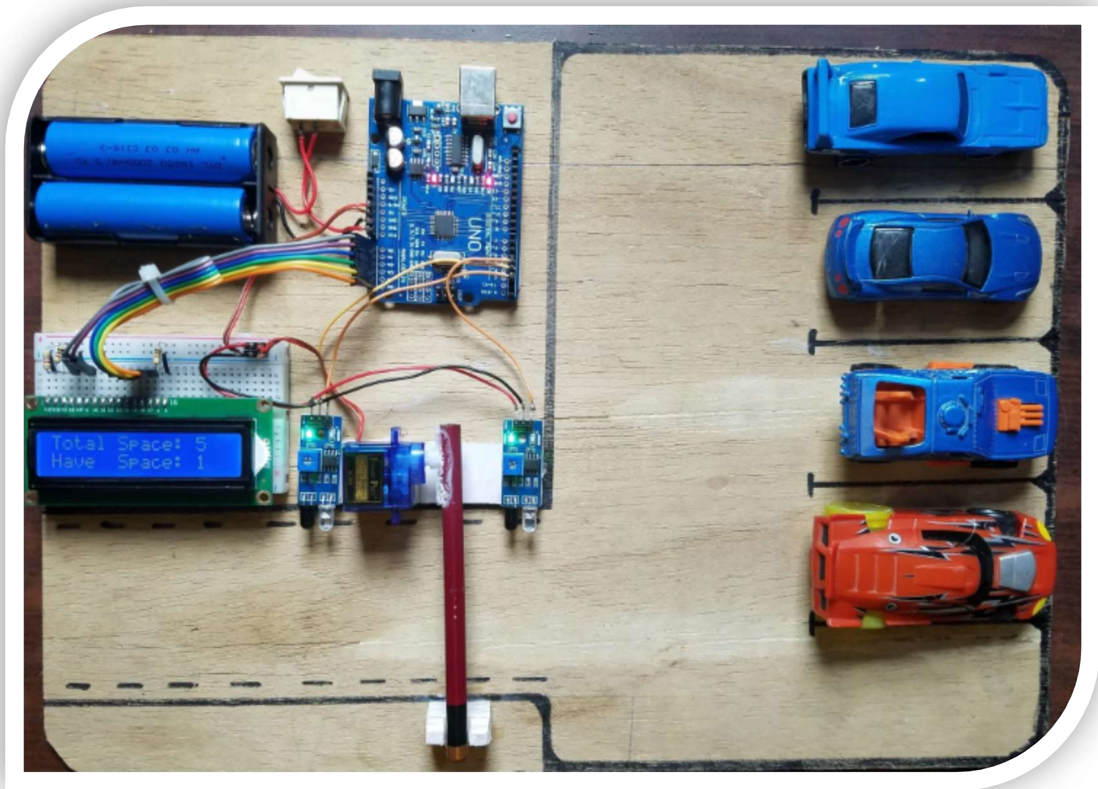
Software Required

- Arduino IDE and WOKWI project simulation



- The Signal pins of both IR Sensors are connected to the Arduino Digital Pins 2 and 4.
- The Signal Pin of the Servo is connected to the digital pin 3.
- Connect the positive terminal of the power supply to VIN on the Arduino and the negative terminal to GND.
- This completes the circuit diagram for the car parking system.

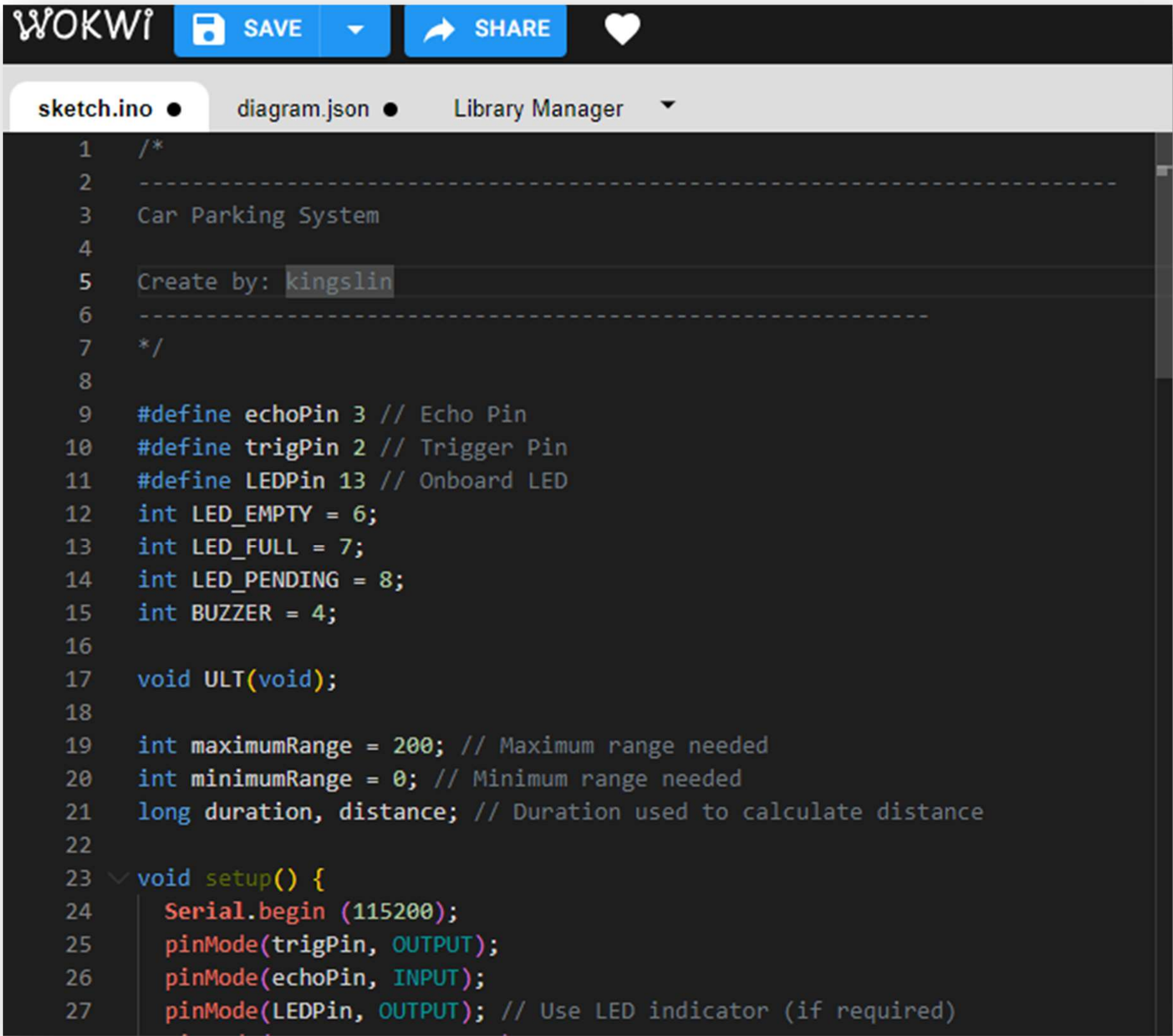
Working Principle



- The working concept of this involves 4 components: IR Sensor, Arduino board, Servo motors, and the LCD Display.
- The IR sensors are continuously scanning both sides of the crossing for cars so they can give an alert when the car is either coming or leaving.
- As soon as the car approaches a crossing from either side the command is sent to the Arduino board. The Arduino board upon receiving the command gives out the signal to the servo to open the crossing.
- The Arduino then gives out the command to LCD Display to either increase or decrease the number of empty spaces.
- The whole process gets started again. This completes the working concept of this project.

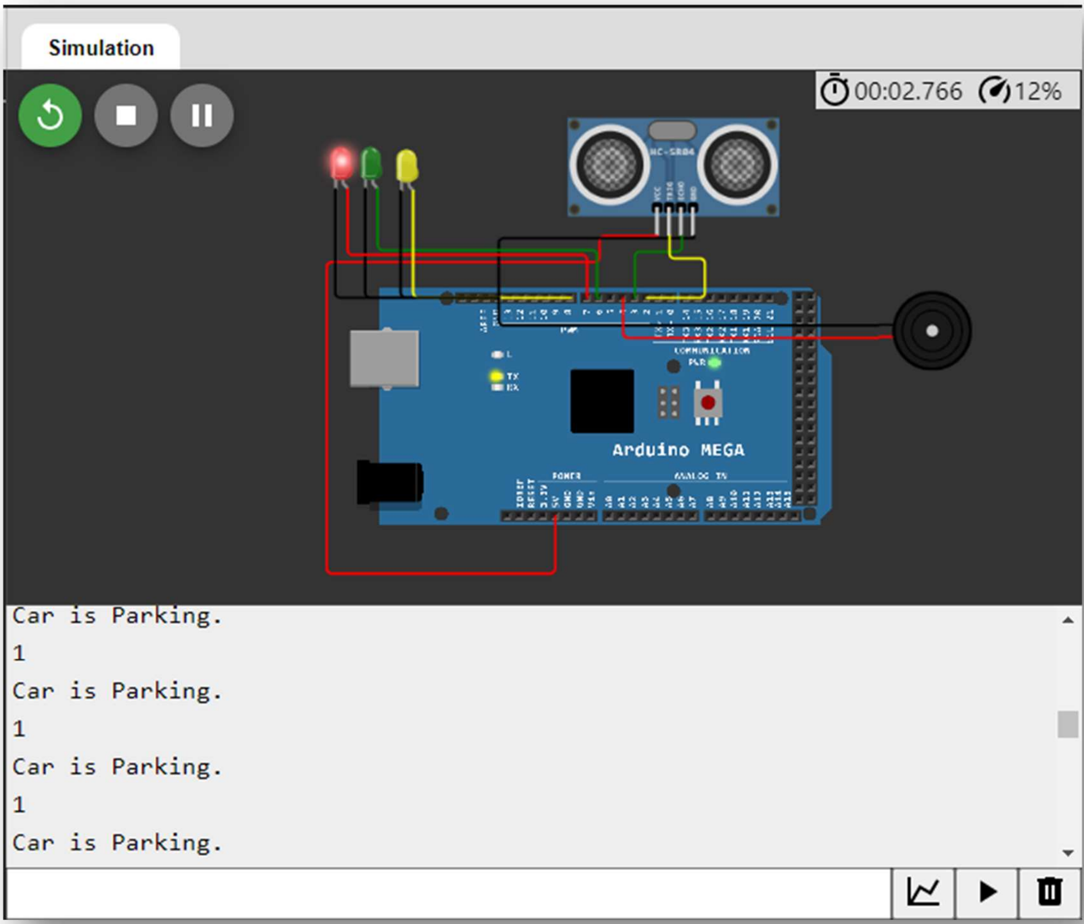
Arduino Code & Wokwi testing code

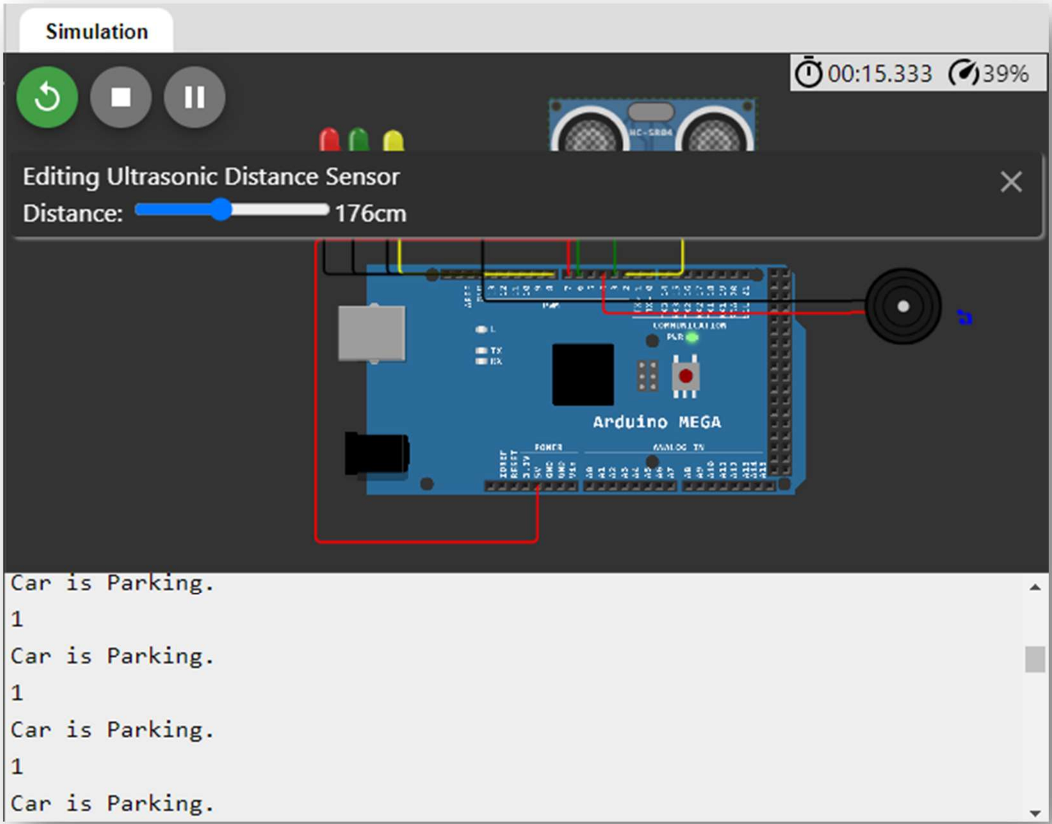
Screen photos

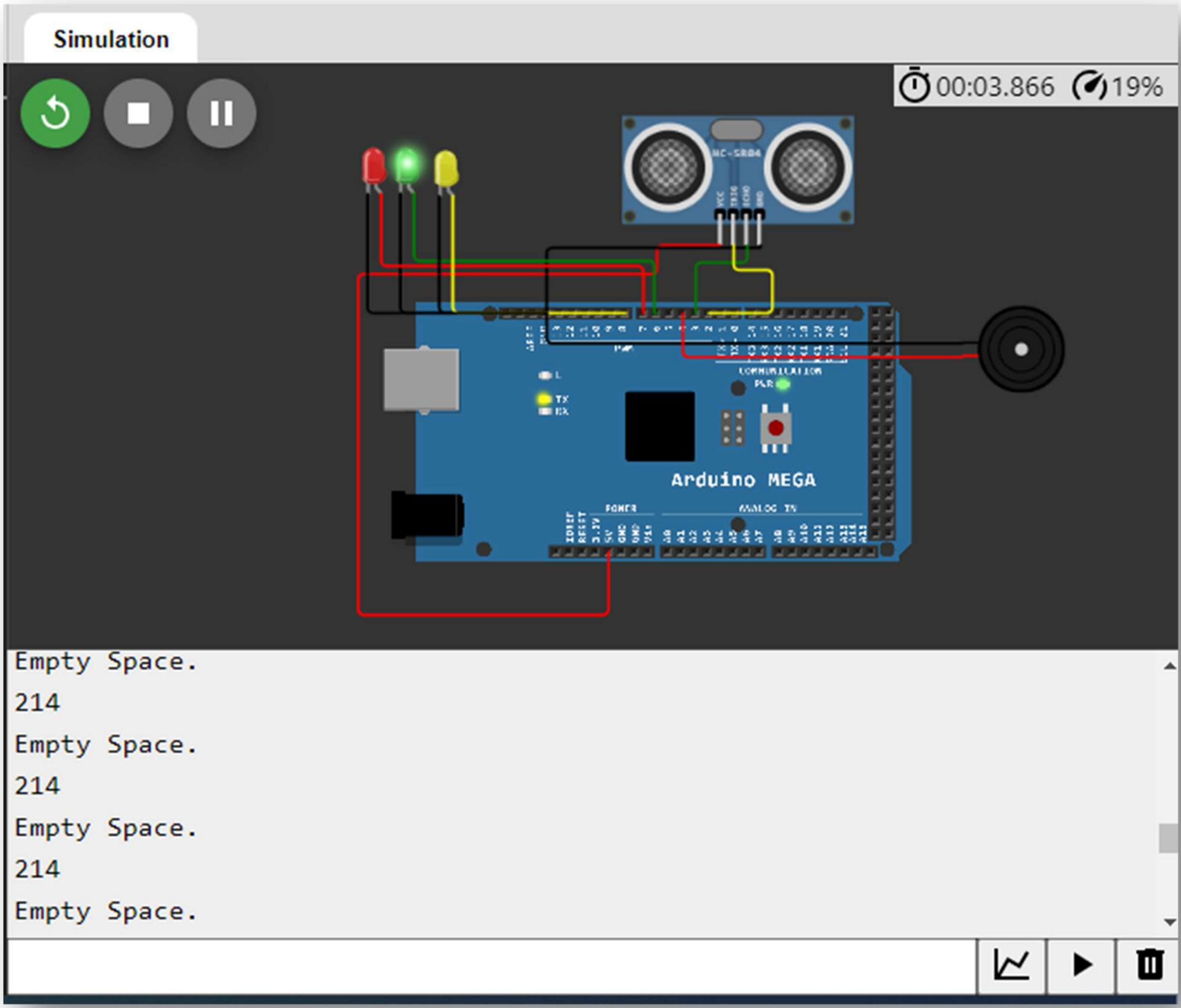


```
  pinMode(LED_EMPTY, OUTPUT);
  pinMode(LED_FULL, OUTPUT);
  pinMode(LED_PENDING, OUTPUT);
  pinMode(BUZZER, OUTPUT);
}

void loop() {
  ULT();
  Serial.println(distance);  //show distance

  /*vacant, out green light*/
  if(distance >= 200){
    digitalWrite(LED_EMPTY,1);
    digitalWrite(LED_PENDING,0);
    digitalWrite(LED_FULL,0);
    Serial.println("Empty Space.");
  }

  /*someone is parking, out yelow light*/
  else if(distance < 200 && distance >= 50){
    digitalWrite(LED_EMPTY,0);
    digitalWrite(LED_PENDING,1);
    digitalWrite(LED_FULL,0);
    tone(BUZZER, 800);
    delay(100);
    digitalWrite(LED_EMPTY,0);
    digitalWrite(LED_PENDING,0);
```

```
    noTone(BUZZER);
    delay(500);
    Serial.println("Car is going to park here or going out.");
  }

  /*occupied, out red light*/
    else{
    digitalWrite(LED_EMPTY,0);
    digitalWrite(LED_PENDING,0);
    digitalWrite(LED_FULL,1);
    Serial.println("Car is Parking.");
  }
}

void ULT(){
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);

  //Calculate the distance (in cm) based on the speed of sound.
  distance = duration / 58.2;
}
```


Simulation
00:02.766
12%
Arduino MEGA
Car is Parking.
1
Car is Parking.
1
Car is Parking.
1
Car is Parking.


Simulation
00:15.333
39%
Editing Ultrasonic Distance Sensor
Distance:
176cm
Arduino MEGA
Car is Parking.
1
Car is Parking.
1
Car is Parking.
1
Car is Parking.


Simulation
00:03.866
19%
Arduino MEGA
Empty Space.
214
Empty Space.
214
Empty Space.
214
Empty Space.


CAR PARKING SYSTEM

The feature Benefits of Smart Parking Technology

- Optimized parking.
- Reduced traffic.
- Reduced pollution.
- Enhanced User Experience.
- Integrated Payments and POS.
- Increased Safety.
- Real-Time Data and Trend Insight.
- Decreased Management Costs.

Project source code(demo project checking code:)

```
const int trigPin1 = 27;
const int echoPin1 = 26;

const int trigPin2 = 2;
const int echoPin2 = 15;

const int trigPin3 = 18;
const int echoPin3 = 5;

const int ledPin1 = 13;
const int ledPin2 = 12;
const int ledPin3 = 14;

long duration;
int distance;

void setup () {
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);

  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);

  pinMode(trigPin3, OUTPUT);
```

```
  pinMode(echoPin3, INPUT);

   pinMode(ledPin1, OUTPUT);
   pinMode(ledPin2, OUTPUT);
   pinMode(ledPin3, OUTPUT);

  Serial.begin(9600);
}
void loop() {
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);

  duration = pulseIn(echoPin1, HIGH);

  distance = duration * 0.034 / 2;

if (distance < 200) {
  digitalWrite(ledPin1, LOW);
  Serial.println("Parking Space 1 : Occupied");
}else {
  digitalWrite(ledPin1, HIGH);
  Serial.println("Parking Space1 : Vacant");
}
delay(1000);

  Serial.print("Distance");
  Serial.println(distance);

   digitalWrite(trigPin2, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin2, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin2, LOW);

  duration = pulseIn(echoPin2, HIGH);

  distance = duration * 0.034 / 2;

if (distance < 200) {
  digitalWrite(ledPin2, LOW);
  Serial.println("Parking Space 2 : Occupied");
}else {
  digitalWrite(ledPin2, HIGH);
  Serial.println("Parking Space2 : Vacant");
}
delay(1000);

  Serial.print("Distance");
  Serial.println(distance);
```

```
  digitalWrite(trigPin3, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin3, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin3, LOW);

  duration = pulseIn(echoPin3, HIGH);

  distance = duration * 0.034 / 2;

if (distance < 200) {
  digitalWrite(ledPin3, LOW);
  Serial.println("Parking Space 3 : Occupied");
}else {
  digitalWrite(ledPin3, HIGH);
  Serial.println("Parking Space3 : Vacant");
}
delay(1000);

  Serial.print("Distance");
  Serial.println(distance);
}
```

DEMO OUTPUT: