|  |
| --- |
|  |
| SMART PARKING Using IoT |
| WhatsApp Image 2023-10-29 at 2.01.47 PM.jpeg |

|  |
| --- |
| Submitted by,    KAVIN E - 812921106020  SRIVARUDAN J - 812921106033  KAVIYARASAN A - 812921106021  SANTHOSH P - 812921106029 |

TABLE OF CONTENTS

* Abstract
* Objectives
* Introduction
* Block diagram
* Explanation
* System architecture
* Proposed system
* Hardware
* Software
* Schmit diagram
* Components
* GSM module
* IR sensor
* RFID card
* READER module
* SERVO motor
* Arduino Nano
* WIFI module
* Program
* Output
* Advantage
* Disadvantage
* conclusion

Abstract

The project involves integrating IoT sensors into public transportation vehicles to monitor ridership, track locations, and predict arrival times. The goal is to provide real-time transit information to the public through a public platform, enhancing the efficiency and quality of public transportation services. an effective way of finding empty spaces and managing the number of vehicles moving in and out in complex multi storeyed parking structures by detecting a vehicle using IR sensors and thus providing a feedback. The fully automated smart car parking system is rudimental and does not require heavy lines of code nor expensive equipment. It is a simple circuit built for the exact need of purpose. This automated system is used to find the vacancy in parking spaces available and navigate the driver to reach the desired space using visuals and in an effective manner, thus reducing search time. This system is required for malls, multistore parking structures, IT hubs and parking facilities. This makes sure the requirement of labour is insubstantial. This project includes defining objectives, designing the IoT sensor system, developing the real-time transit information platform, and integrating them using IoT technology and Python.

Objective

The basic objective of a smart parking solution is to identify a vehicle’s presence or absence in a particular parking space with a high degree of accuracy, and to pass on this data into a system for visualization and analysis – to be available for parking asset managers and/or enforcement officers.

Keeping in mind the objectives mentioned above, the next step is to take into consideration following important features:

* Accuracy of detecting a vehicle presence/absence
* Total cost of solution
* Privacy concerns

The laser scanner sensors are well known for their accuracy of detecting a vehicle presence – therefore the sensor located at the entrance and exit of a parking area will count with high accuracy the entrance and exit of the vehicles, taking into consideration even two cars stopping very close to each other.

The total cost of solution considering the initial purchasing and installation cost, you will have low maintenance cost and no need of replacing batteries.

The sensors will just count the vehicles without recording any data. The system is not based on cameras, but again on sensors and eventually a display for counting the cars. Therefore, the privacy is totally granted.Sensor solution is a reliable, and cost-effective modern smart parking solution.

INTRODUCTION

Internet of thing (IoT) has the ability to transfer data through network without involving human interactions. IoT allows user to use affordable wireless technology and also helps the user to transfer the data into the cloud. IoT helps the user tomaintain transparency. The idea of IoT started with theidentity of things for connecting various devices. These devices can be controlled or monitored through computers over internet. IoT contains two prominent words “Internet”and “Things”,

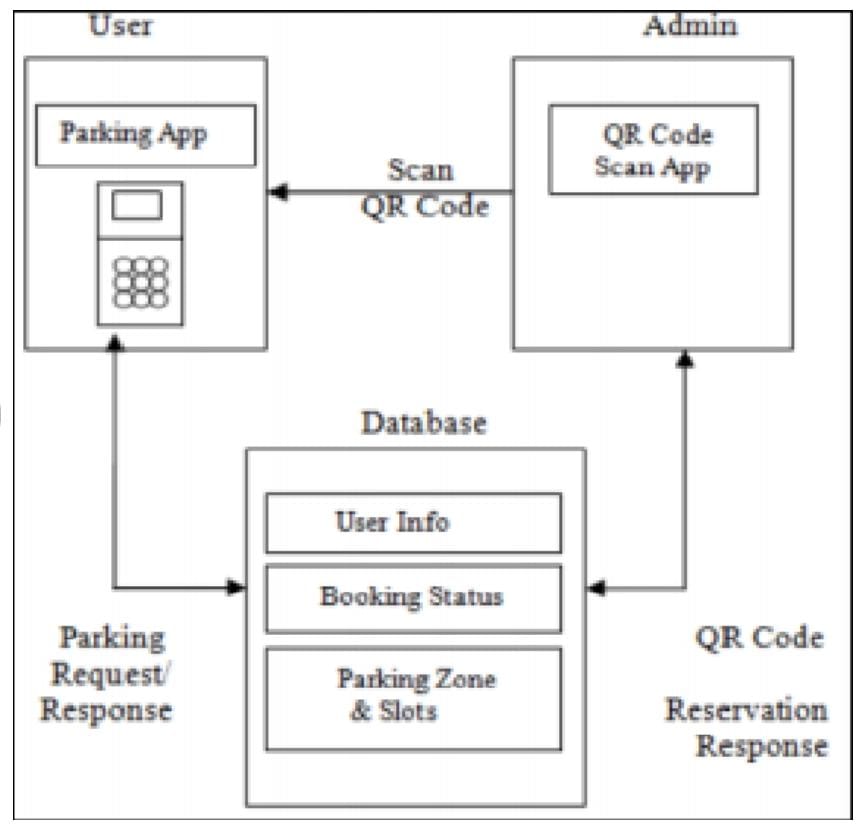
According to the recent survey, there will be a rapid increase in the vehicle’s population of over 1.6 billion around 2035 [7]. Around one million barrels of world’s oil is being burnt everyday [4]. Thus, smart parking system is the key solution to reduce the waste stage of the fuel. The solution for the problems that is being raised.

In the present scenario around us we see excess vehicles and the ineffectiveness to manage them in the correct order. As the population increases day by day the rate of utilization also increases and coping up with the numbers becomes a task. An omnipresent problem around the world is finding a parking space to park your vehicle.

This task looks simple on side roads and interior lanes but the actual problem arises when parking in malls, multistorey parking structures, IT hubs and parking facilities where several hundred cars are parked and it becomes arduous to find a spot. The general approach to finding a parking space is to go around and drive aimlessly until a free space is found.

Finding a parking space could be the easiest task or could be the most tedious one when it involves wide acres of distributed space across one level or multiple levels. The time and fuel are consumed unnecessarily because the destination is unknown. The easiest way of approach is to provide a destination specific driving within the parking structure. A smart car parking system gives a visual output indicating an available parking space rather than driving aimlessly. The driver looks up to the row of LED lights and their colour to deduct a result of determining the parking space availability.

The two main colours used are red and yellow stating occupied and free respectively. These lights are placed at the ceiling of each parking space and the driver looks up and follows the set of LEDs and searches for a Yellow one.

Block diagram & Explanation

In the past, there have been many works done on smart parking system approaching an even smarter system in where researches have been done and still being done to create a system which is not technologically savvy but also at ease. This paper proposes a design of smart parking system where it helps the users to reserve parking slots using Android application. This project is aimed to create a system that helps people with personal vehicles to find for parking easily at selected areas. Both software and hardware platform have been developed in this system.

Finding a free parking lot in a congested city like Dhaka is very hard. Here, if anyone wants to go outside from home with personal car first thing comes in his mind is about parking, where he will park his car. Most of the cases, people go to a parking station and find that all parking slot are full and then he have to search for another parking lot. So, it is a big hassle and many people keep in fear about parking of his car when he gets out with his car.

I realized that, to enjoy a better transport a better parking system is necessary especially in a congested city like Dhaka. So, I was thinking, how the problem can be solved and finally I succeeded to make a cloud based smart parking system and I hope implementing the system can remove the parking problem of my city. ARTIK Cloud is really a nice and appropriate platform for such job. Using this system a user will be will able to find an available parking lot easily using mobile or web app from anywhere. I also used Intel Edison with a display which may be placed several important locations of the city or road from where it will be possible to find free parking lot. The system updates parking data every 30 seconds. In this project I will show you how you can easily build such smart system. I will use ARTIK Cloud platform, a cool IoT cloud platform

One of the most important problems facing large cities is congestion and parking . So, using Automated Parking System Management is an efficient technique using the Internet of Things to manage the garage . Smart parking is an electronic tool that enables the user to find vacant parking spaces through information technology and by using appropriate sensors . Among the most used types in smart parking, systems are data routing systems, smart payment systems, and electronic car parks. These types require disclosure of whether parking spaces are vacant or not. With the user registration in the system, a unique identifier is created for him, and with the booking, it has the booking details, and via their smartphones, the entire time, exit time, and money are calculated. The System building consists of, the lowest level, including the functions of sensing, data transmission is created during a middle level, and upper-level deals with the storage and processing information, and user interfaces .

SYSTEM ARCHITECTURE

## Proposed system

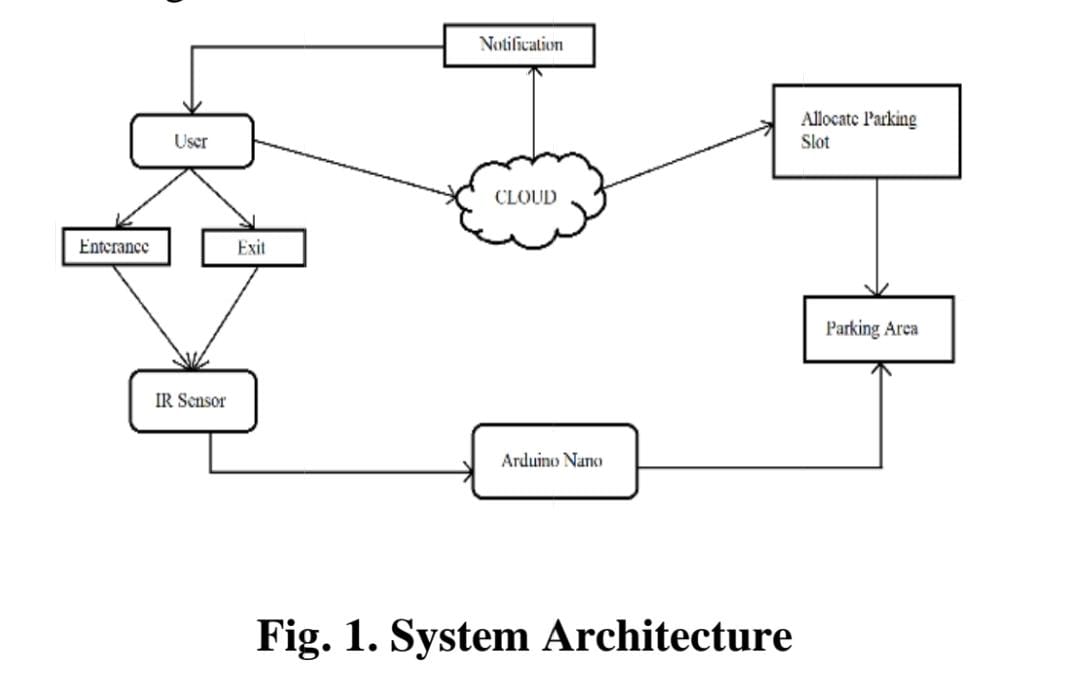
## hardware

## software

## A. Proposed System

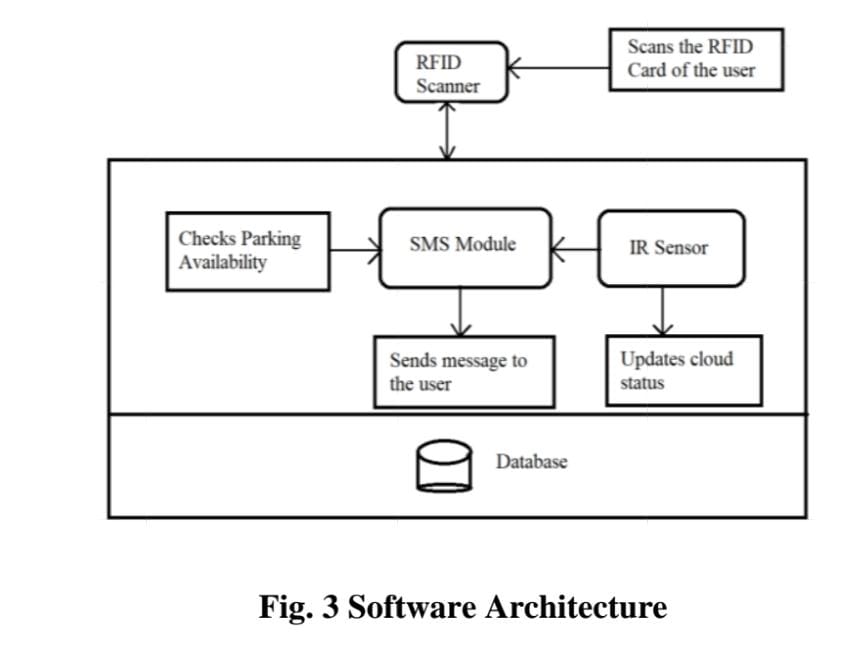
It consists of three sections: first section is the parking area which includes Arduino devices along with IR sensor. The user interacts with the parking area with the help of these devices. The user cannot enter the parking area without the help of RFID card.

The second section contains the cloud based web services which acts a mediator between the user and parking area. The cloud is updated depending upon the availability of the parking area. The admin administers the cloud services and it can also be viewed by the user for checking the availability. The third section is the user side. The user gets notification on the basis of the availability via SMS through GSM module.

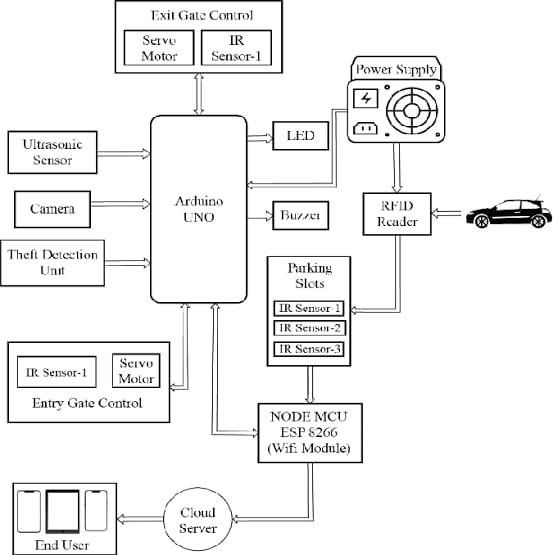
 The ISPS can help the users to search the unoccupied spaces in the user-specific area within a very short time and also can suggest the nearest location from user destination. Additionally, ISPS supports the users with automated management of parking spaces and cares about fuel utilization.

# Hardware

# WhatsApp Image 2023-10-25 at 3.08.20 PM.jpeg The three main hardware components used are GMS module, RFID card, IR sensors. A user is allowed inside a parking space only if the user has a RFID card. RFID card contains the information of the registered user. As the car enters the parking slot, reader module scans the registered user’s RFID tag. The data is sent to the ardunio for checking the availability of the car parking and simultaneously, the user is notified through SMS about the status of the parking area. The GSM module sends the message according to the availability. IR sensor sends the signals according to the presence of the vehicleC. Software

 The cloud server acts as a mediator between the modules. The cloud server is connected to the Wi-Fi module. The user receives messagesthrough the SMS module while the car enters and exits the parking area using RFID card. The messages sent by the SMS module are managed by the cloud. As soon as the IR sensor detects the car, the status of the cloud will be updated from 0 to 1 and when the car leaves the parking area the status of the car will be updated from 0 to 1.

Schmit diagram



Components

* GSM module
* IR sensor
* RFID card
* READER module
* SERVO motor
* Arduino Nano
* WIFI module

## GSM Module

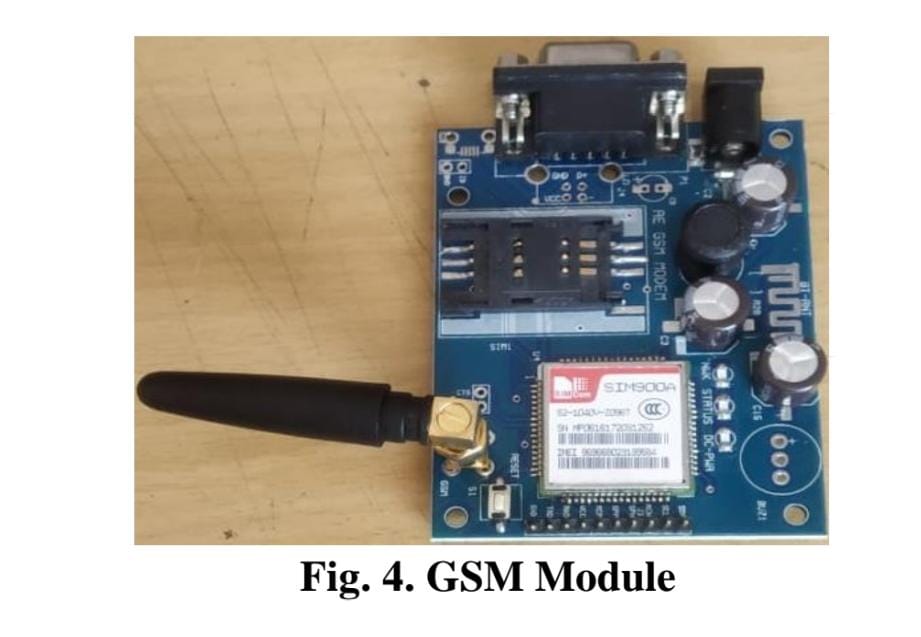
The GSM module is a circuit which is used to setucommunication between mobile phones and microcontroller. It is used to send SMS, MMS and voice messages through mobile network. GPRS extension in GSM allows high data transmission. GSM uses time division multiple access approach for transmission.

A GSM module or a GPRS module is a chip or circuit that will be used to establish communication between a mobile device or a computing machine and a GSM or GPRS system. The modem (modulator-demodulator) is a critical part here.

* SIM900 GSM Module
* SIM900 GSM Module

These modules consist of a GSM module or GPRS modem powered by a power supply circuit and communication interfaces (like RS-232, USB 2.0, and others) for computers.

A GSM modem can be a dedicated modem device with a serial, USB, or Bluetooth connection, or it can be a mobile phone that provides GSM modem capabilities.

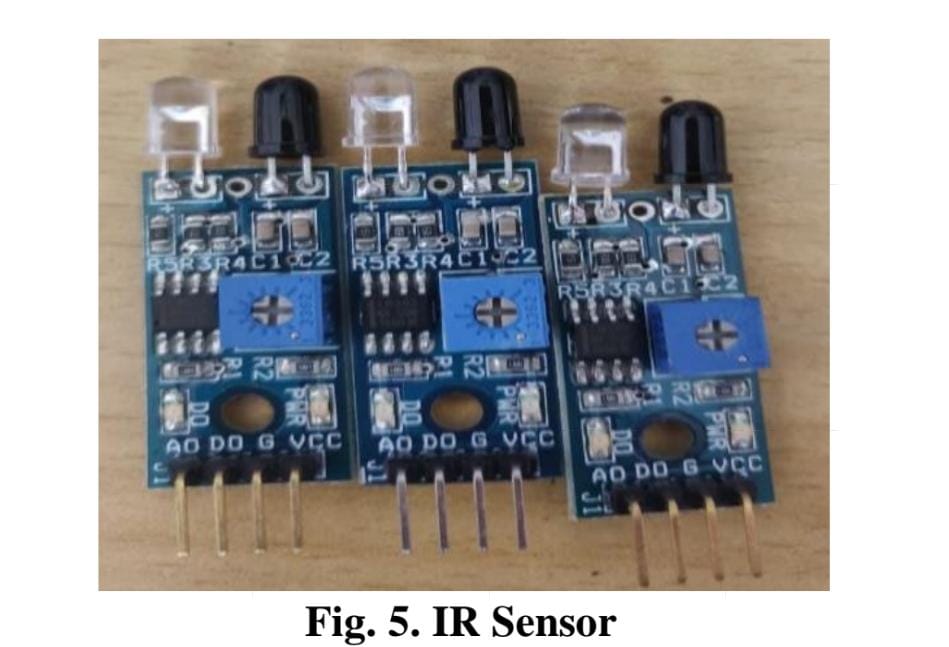


## IR Sensor

An infrared sensor is basically an electronic device which is used to detect the presence of objects. Infrared light is emitted by this device. If this device does not detect any IR light reflected back that means there is no object present. If the light is detected by the sensor there is an object present.

The emitter is simply an IR LED (Light Emitting Diode) and the detector is simply an IR photodiode . Photodiode is sensitive to IR light of the same wavelength which is emitted by the IR LED. When IR light falls on the photodiode, the resistances and the output voltages will change in proportion to the magnitude of the IR light received.

There are five basic elements used in a typical infrared detection system: an infrared source, a transmission medium, optical component, infrared detectors or receivers and signal processing. Infrared lasers and Infrared LED’s of specific wavelength used as infrared sources.

 The three main types of media used for infrared transmission are vacuum, atmosphere and optical fibers. Optical components are used to focus the infrared radiation or to limit the spectral response.

## RFID Card

RFID (Radio Frequency Identification) is a wireless communication technology that uses radio waves to automatically identify tagged objects or things. It transmits data from an RFID tag to an RFID reader using an antenna, enabling accurate and real-time tracking.

In this technology, the RFID reader and RFID tag are the two basic components used, where data is digitally encoded in an RFID tag that can be read by the RFID reader.

The RFID reader is a device that has multiple antennas that transmit radio waves and receive radio signals back from the RFID tag. The RFID tag uses radio frequency to communicate its information, including its own identity, to nearby readers.

RFID technology is capable of storing, recovering, and re-recording a vast amount of data… RFID tags are made up of integrated circuit (IC), an antenna, and a substrate. It is an identification badge or credit card that transfers its contents about an object to the Reader module. RFID tag transfers data about an object through radio waves.When RFID tags are attached to devices they can also be used for tracking.



## READER Module

This module is a device which scans and gathers the information from the RFID Card. This card can be used to track objects. As the car enters the parking area, the user scans the RFID card and all the information stored in card is transferred to the admin through this module.

EM-18 is one of the commonly used modules for Radio Frequency Identification Projects. It can be directly interfaced with a microcontroller using UART communication and with a PC using an RS232 converter. This module can work with any 125KHz RFID tags and it can provide UART / Wiegand26 output formats.

EM18 RFID Serial TTL Module is an 125KHz RFID Reader with built in power supply and can be interfaced with RS232 and TTL with Arduino and raspberry pi.

 RFID Reader Modules are an unfinished unit needing memory, a power supply, an antenna, a processor, and a case. The characteristics are type, frequency, interface, and package/case.

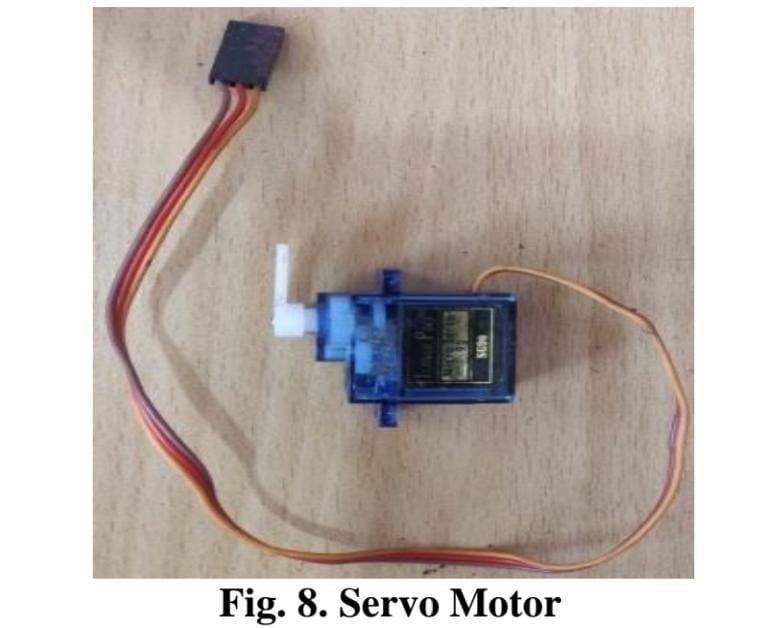
## Servo Motor

It is a rotator device that allows the control of angular as well as linear motion. A servo motor is used for the opening and closing of the gate. Servo drivetransmits electrical signals to the servo motor for producing motion.

A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a servo mechanism. If motor is powered by a DC power supply then it is called DC servo motor, and if it is AC-powered motor then it is called AC servo motor.

For this tutorial, we will be discussing only about the DC servo motor working. Apart from these major classifications, there are many other types of servo motors based on the type of gear arrangement and operating characteristics. A servo motor usually comes with a gear arrangement that allows us to get a very high torque servo motor in small and lightweight packages. Due to these features, they are being used in many applications like toy car, RC helicopters and planes, Robotics, etc.

Servo motors are rated in kg/cm (kilogram per centimeter) most hobby servo motors are rated at 3kg/cm or 6kg/cm or 12kg/cm. This kg/cm tells you how much weight your servo motor can lift at a particular distance. For example: A 6kg/cm Servo motor should be able to lift 6kg if the load is suspended 1cm away from the motors shaft, the greater the distance the lesser the weight carrying capacity. The position of a servo motor is decided by electrical pulse and its circuitry is placed beside the motor.



## Arduino Nano

It is a compact board which can be used in various devices and various field. It has overall 22 input/output pins out of which 14 pins are digital pins. It has a flash memory of

about 32 kb. These pins can control the operations of digital pins as well as analogy pins. This module is a breadboard friendly board which can be easily used anywhere.

The Arduino Nano is Arduino's classic breadboard friendly designed board with the smallest dimensions. The Arduino Nano comes with pin headers that allow for an easy attachment onto a breadboard and features a Mini-B USB connector.

Arduino Nano is one type of microcontroller board, and it is designed by Arduino.cc. It can be built with a microcontroller like Atmega328. This microcontroller is also used in Arduino UNO. It is a small size board and also flexible with a wide variety of applications. Other Arduino boards mainly include Arduino Mega, Arduino Pro Mini, Arduino UNO, Arduino YUN, Arduino Lilypad, Arduino Leonardo, and Arduino Due. And other development boards are AVR Development Board, PIC Development Board, Raspberry Pi, Intel Edison, MSP430 Launchpad, and ESP32 board.

This board has many functions and features like an Arduino Duemilanove board. However, this Nano board is different in packaging. It doesn’t have any DC jack so that the power supply can be given using a small.



## WIFI Module

It is used to send data from embedded system to the internet using URL by HTTP POST method using TCP/IP protocol. It is developed by espressif systems. It is a 32 bit

microcontroller with 80kb user data. It contains 16 gpio pins.

ESP8266 is the most widely used Wi-Fi module. It is a low-cost microchip with a full TCP/IP stack and microcontroller capability, produced by Espressif Systems. This small module allows microcontrollers to connect to a Wi-Fi network and make simple TCP/IP connections.

ESP8226 comes with the capabilites of:-

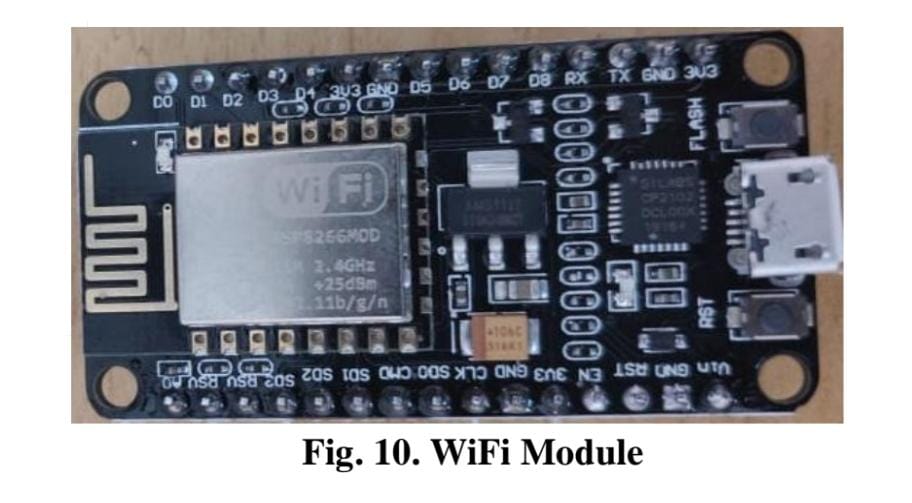
1. 2.4 Ghz Wi-Fi

2. General-purpose input/output (16 GPIO)

3. Inter-Integrated Circuit (I²C) serial communication protocol

4. Analog-to-digital conversion (10-bit ADC)

It runs at operating voltage of 3V and can handle maximum voltage of around 3.6V. It can be easily interfaced with microcontrollers board via Serial Port. There are numerous breakout boards available based on ESP8266 Wifi Module like ESP8266 NodeMCU V3. Because of its compact size, its most importantly used in autonomous project.



Program

def init(self, v\_type, v\_number):

self.v\_type = v\_type

self.v\_number = v\_number

self.vehicle\_types = {1: 'c', 2: 'b', 3: 't'}

def str(self):

return self.vehicle\_types[self.v\_type]

class Slot:

def init(self):

self.vehicle = None

@property

def is\_empty(self):

return self.vehicle is None

class Parking:

def init(self, rows, columns):

self.rows = rows

self.columns = columns

self.slots = self.\_get\_slots(rows, columns)

def start(self):

while True:

try:

print(options\_message)

option = input("Enter your choice: ")

if option == '1':

self.\_park\_vehicle()

if option == '2':

self.\_remove\_vehicle()

if option == '3':

self.show\_layout()

if option == '4':

break

except ValueError as e:

print(colored(f"An error occurred: {e}. Try again.", "red"))

print(colored("Thanks for using our parking assistance system", "green"))

def \_park\_vehicle(self):

vehicle\_type = self.\_get\_safe\_int("Available vehicle types: 1. Car\t2. Bike\t3. Truck.\nEnter your choice: ")

if vehicle\_type not in [1, 2, 3]:

raise ValueError("Invalid vehicle type specified")

vehicle\_number = input("Enter vehicle name plate: ")

if not vehicle\_number:

raise ValueError("Vehicle name plate cannot be empty.")

vehicle = Vehicle(vehicle\_type, vehicle\_number)

print('\n')

print(colored(f"Slots available: {self.\_get\_slot\_count()}\n", "yellow"))

self.show\_layout()

print('\n')

col = self.\_get\_safe\_int("Enter the column where you want to park the vehicle: ")

if col <= 0 or col > self.columns:

raise ValueError("Invalid row or column number specified")

row = self.\_get\_safe\_int("Enter the row where you want to park the vehicle: ")

if row <= 0 or row > self.rows:

raise ValueError("Invalid row number specified")

slot = self.slots[row-1][col-1]

if not slot.is\_empty:

raise ValueError("Slot is not empty. Please choose an empty slot.")

slot.vehicle = vehicle

def \_remove\_vehicle(self):

vehicle\_number = input("Enter the vehicle number that needs to be removed from parking slot: ")

if not vehicle\_number:

raise ValueError("Vehicle number is required.")

for row in self.slots:

for slot in row:

if slot.vehicle and slot.vehicle.v\_number.lower() == vehicle\_number.lower():

vehicle: Vehicle = slot.vehicle

slot.vehicle = None

print(colored(f"Vehicle with number '{vehicle.v\_number}' removed from parking", "green"))

return

else:

raise ValueError("Vehicle not found.")

def show\_layout(self):

col\_info = [f'<{col}>' for col in range(1, self.columns + 1)]

print(colored(f"|{''.join(col\_info)}|columns", "yellow"))

self.\_print\_border(text="rows")

for i, row in enumerate(self.slots, 1):

string\_to\_printed = "|"

for j, col in enumerate(row, 1):

string\_to\_printed += colored(f"[{col.vehicle if col.vehicle else ' '}]",

"red" if col.vehicle else "green")

string\_to\_printed += colored(f"|<{i}>", "cyan")

print(string\_to\_printed)

self.\_print\_border()

def \_print\_border(self, text=""):

print(colored(f"|{'-' \* self.columns \* 3}|{colored(text, 'cyan')}", "blue"))

def \_get\_slot\_count(self):

count = 0

for row in self.slots:

for slot in row:

if slot.is\_empty:

count += 1

return count

@staticmethod

def \_get\_slots(rows, columns):

slots = []

for row in range(0, rows):

col\_slot = []

for col in range(0, columns):

col\_slot.append(Slot())

slots.append(col\_slot)

return slots

@staticmethod

def \_get\_safe\_int(message):

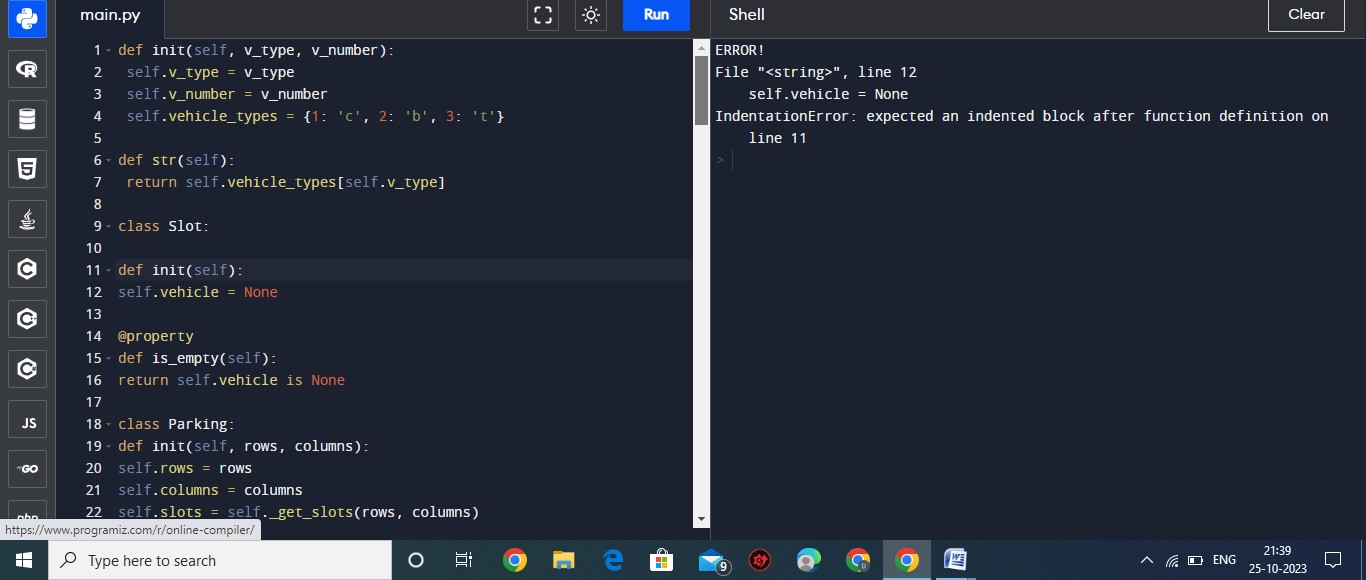
try:

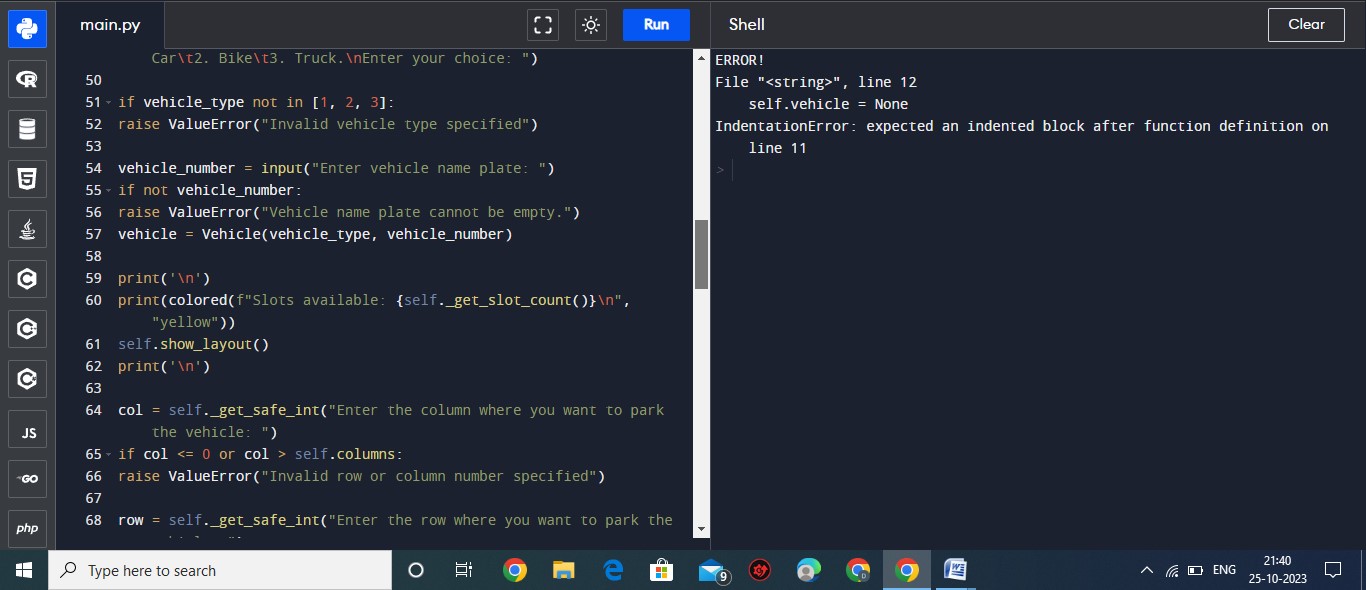
val = int(input(message))

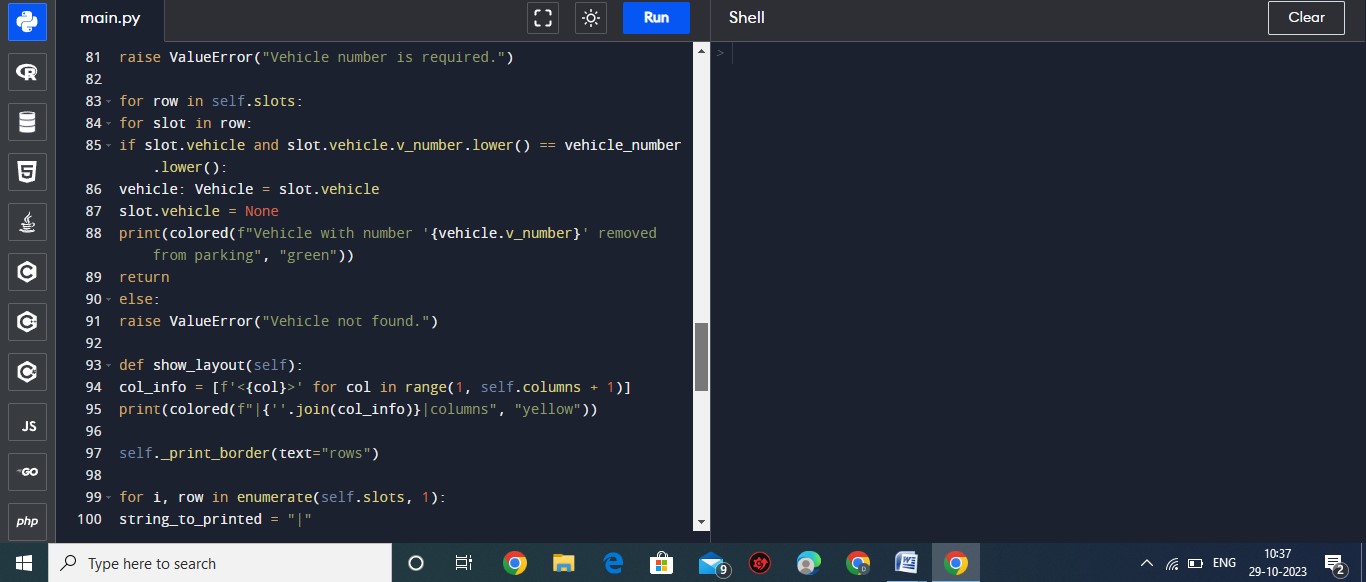
return val

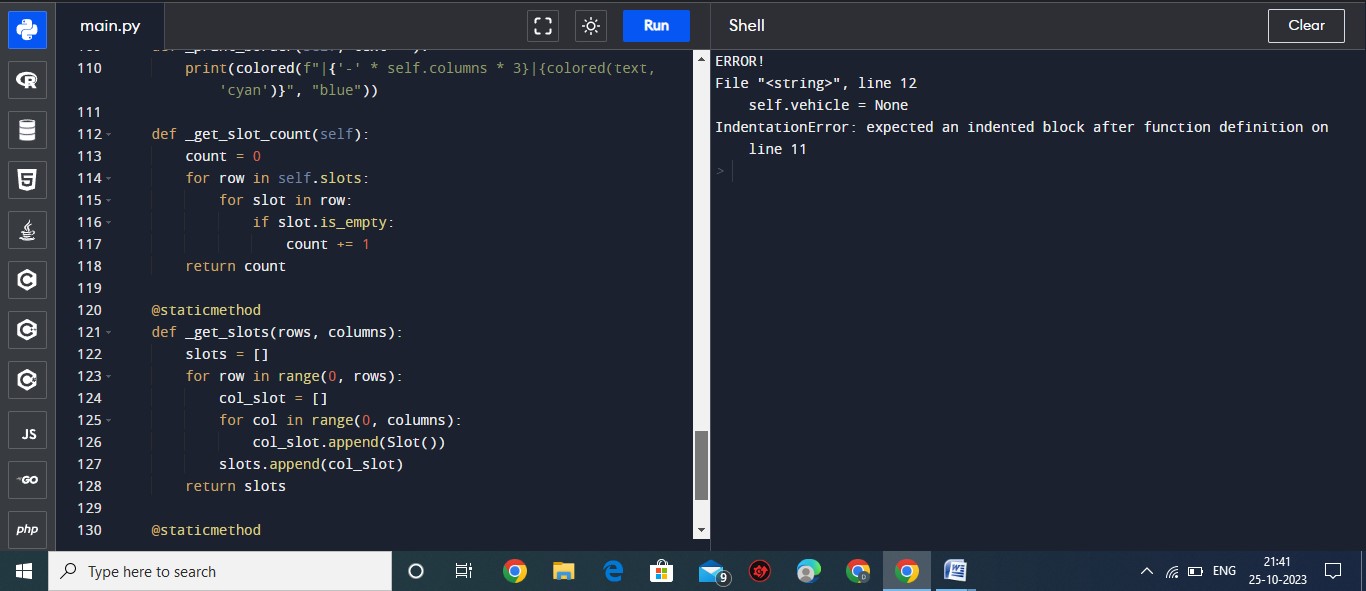
except ValueError:

raise ValueError("Value should be an integer only")

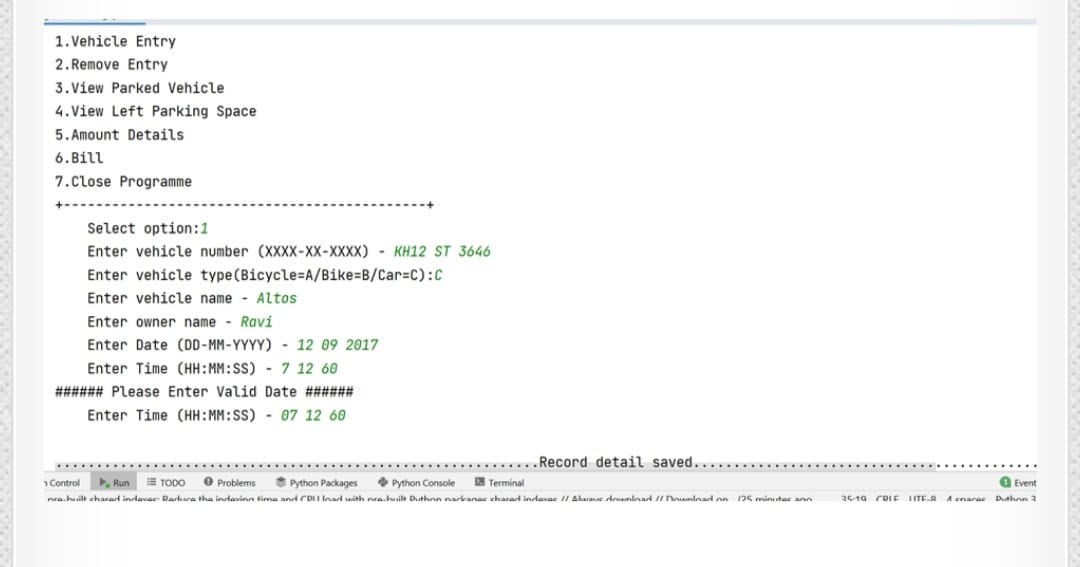
program







output



# Benefits analysis 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.
* Increased Service and Brand Image.

# Disadvantages

* High installation and maintenance cost
* Limited vehicle size accommodation
* System breakdown causes inconvenience
* Dependent on electricity
* Less human employment opportunities

conclusion

The services provided by smart parking have become the essence of building smart cities. This paper focused on implementing an integrated solution for smart parking. The proposed system has several advantages, including detecting parking spaces using the Internet of Things and calculating the time of entry and exit and calculating the expected cost. An attractive and effective application was designed for Android mobile phones. The system benefits from avoiding wasting time and reducing pollution and fuel consumption. Users can book a car park for 24 hours.