

**PROJECT REPORT**

**ON**

**Automated Parking system**

**Submitted by :-**

**Name : ABHISHEK KUMAR**

**Roll No. : MCA/10058/21**

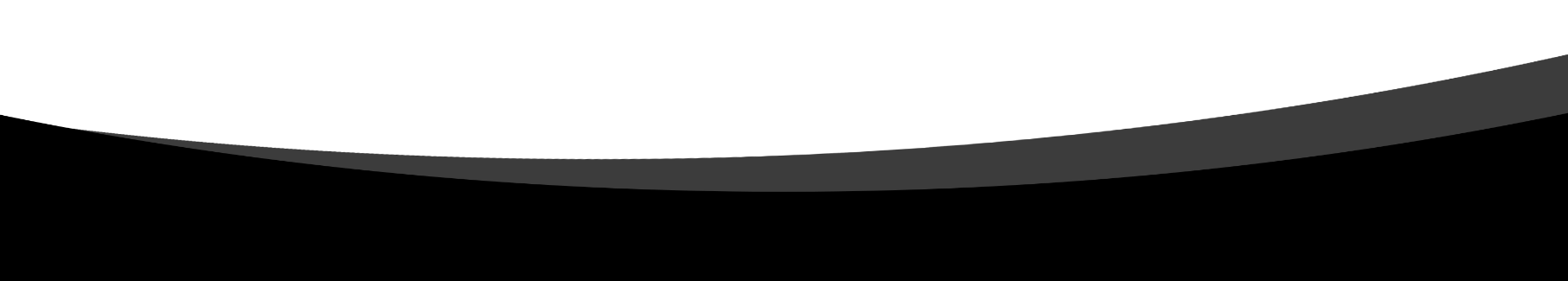
**Department : MCA**

**Session : 2021-2023**

**Under the guidance of**

***Dr. Anup Keshri***

Signature



**INTRODUCTION**

The project entitled **Automated Parking system** is to manage the two wheeler and four wheeler parking. This software Automated Parking System Project in Python is a set of innovative technologies that provide parking sector solutions. Any parking management system’s main concept is self-explanatory: it’s a system that assists individuals, businesses, and organizations in managing their parking spaces.

# Hardware and Software requirement

**Hardware Requirement**

Processor : Intel Core Duo 2.0 GHz or more

RAM : 1 GB or More

Harddisk : 80GB or more

Monitor : 15” CRT, or LCD monitor

Keyboard : Normal or Multimedia

Mouse : Compatible mouse

**Software Requirement**

Front End : PYTHON & PHP edition With Sql Server Compact Edition

Back End : MS Sql Server

Operation System : Windows 8

Or Windows 7

**OBJECTIVE:**

This **Python Project on Automated Parking system** is mostly concerned with dealing with client parking details such as number and slot. The system also allows vehicle owners to enter information such as their contact information, vehicle number, and vehicle category. However, after entering vehicle information, the system creates a reserve slot that lasts until the car leaves. When it comes to parking spaces, the system uses green and red to signify empty and occupied slots, accordingly. The system displays all parked vehicles under the manage vehicles area of the app, where the user can cancel the parking after it is completed.

**Vehicle Automated Parking and Parking History**

Furthermore, the system shows all previous parking history for both two and four-wheelers. Moreover, during the installation, the administrator must specify the total number of parking spaces for both two and four-wheelers. The system, on the other hand, records the current time when a vehicle record for parking is created, and then calculates the total time after the parking records are deleted. The system stores all of these time stamps in the history area, along with information like the customer’s name, contact information, car number, and parking dates.

Finally, for a better user experience when using this Python vehicle parking management system, a clean and basic GUI with simple color combinations is presented. A cross-platform GUI toolkit Qt; PyQT is on board for its UI elements. Presenting a new Python project that includes a user panel with all of the necessary features for follow-up, as well as a competent resource for learning reasons.

**Automated Parking system** **in Python : Features:**

• Add Vehicle Records

• Manage Vehicle Parking

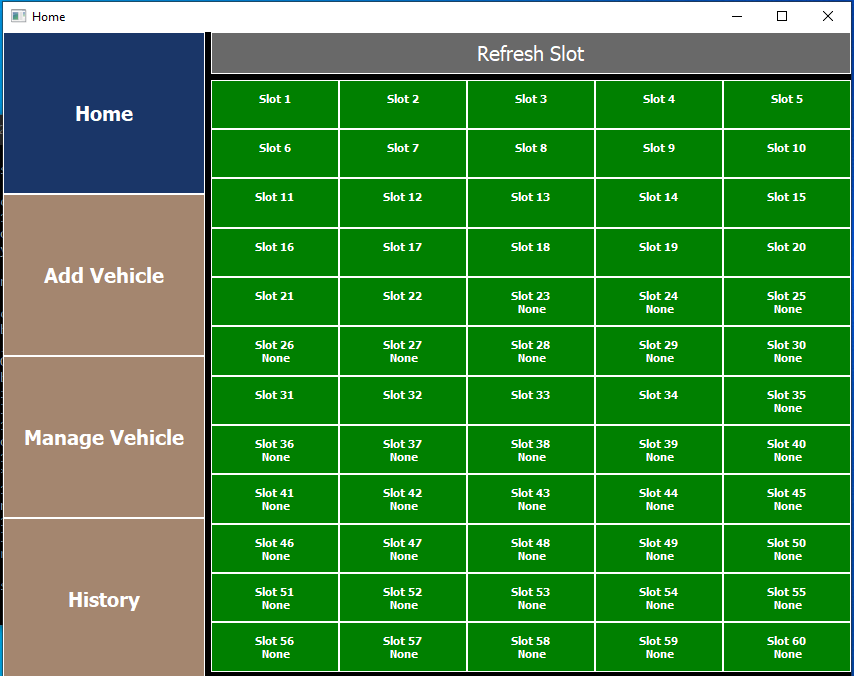
• Available and Empty Parking Indicator

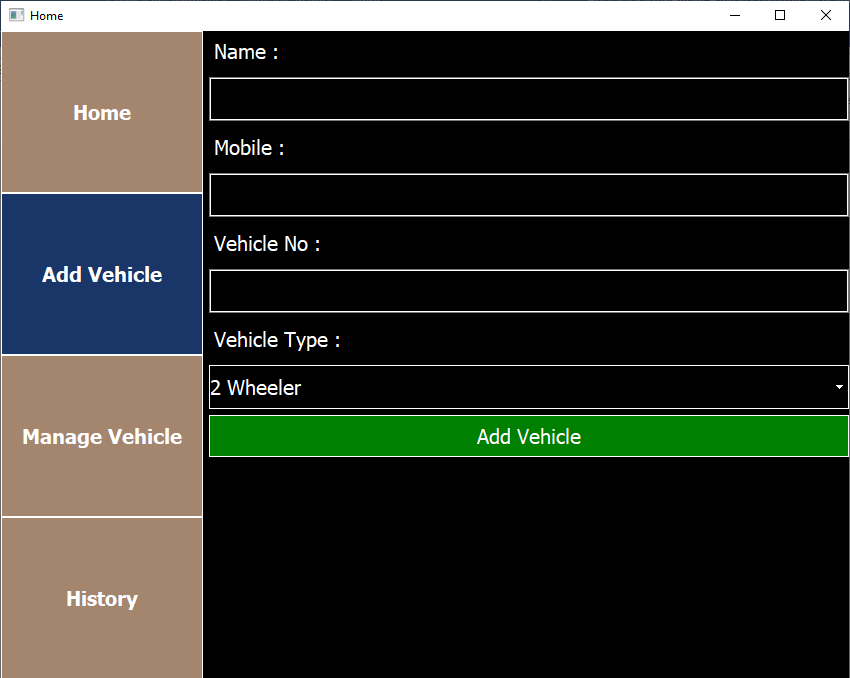
• List parking history

• Vehicle Category

• Total Parking Time Stamp

**SCREENSHOT**

**HOME Page**

**Add Vehicle**