Volume: 14 Issue: III Month of publication: March 2026

DOI: https://doi.org/10.22214/ijraset.2026.77993

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

# Auto Timed Smart Parking Bill Generation System

Dr. V. Ramesh[1], U. Uma[2], E. Harini[3], M. Sandhya[4], G. Anitha[5]
*[1]Associate Professor, [2, 3, 4, 5]UG Students, Department of Electronics & Communication Engineering, Sai Rajeswari Institute of Technology, Proddatur, Andhra Pradesh, 516360.*

***Abstract: Parking management has become a challenging task due to the increasing number of vehicles in urban areas. Traditional parking systems rely on manual monitoring and ticket-based billing, which leads to inefficiency and human errors. This paper presents an Auto Timed Smart Parking Bill Generation System using Optical Character Recognition (OCR) for vehicle license plate detection. The systems use IR sensors, ESP32 microcontroller, servo motors, webcam, and a Flask Server for automated parking management. When a vehicle arrives at the entry gate, the webcam captures the license plate and OCR technology extracts the vehicle number, which is stored along with the entry time in the server database. When the vehicles reach the first exit gate, the system recognizes the license plate again and records the exit time. The total parking duration is calculated automatically the parking bill is generated. A QR code is then displayed for payment. The system also includes a dual exit gate, mechanism to prevent tailgating ensuring secure vehicle exit. The proposed system improves parking efficiency, reduces manual work, and accurate automated billing.***

***Keywords: Smart Parking System, OCR, License Plate Recognition, ESP32, Flask Server, Automated Billing, Anti- Tailgating, QR Code Payment.***

## I. INTRODUCTION

The increasing number of vehicles in urban areas has created challenges in parking management, including congestion, inefficient billing, and security issues. Traditional manual parking system often lead to delays, errors, and unauthorized exits. Smart parking systems using IoT, image processing, and automation can address these problems effectively.[1][2]

The proposed system integrates Optical Character Recognition (OCR) for license plate detection, infrared (IR) sensors for vehicle presence, and servo-controlled gates managed by an ESP32 microcontroller. A Flask-based server records entry and exit times, calculates parking duration, and generates bills. A dual exit gate mechanism prevents tailgating, allowing only one vehicle to exit at a time and enhancing security.

This automated framework reduces manual effort, improves operational efficiency, and provides a secure and vehicle parking management solution.

## II. PROBLEM STATEMENT

Traditional parking systems depend on manual ticket generation and human supervision. This process often leads to delays, inaccurate billing, and inefficient parking management. In addition security issues such as tailgating allow multiple vehicles to exit without proper payment. There is a need for an automated parking system that cam detected vehicles, calculate parking duration automatically, generate bills, and ensure secure exit of vehicles. The proposed system addresses these issues by integrating OCR-based license plate recognition, automated billing, and a dual exit gate mechanism tailgating.

## III. METHODOLOGY

The proposed system operates through a combination of hardware and software components integrated into a centralized parking management framework. At the entry gate a webcam captures the image of the arriving vehicle. The captured image is processed using Optical Character Recognition (OCR) implemented in Python to extract the vehicle license plate number.[3] Once the license plate number is identified, the system records the entry time and stores the data in the server database through a Flask-based web application running on Visual Studio Code.

Infrared sensors installed at the entry gate detectthe presence of a vehicle and trigger the opening of the gate through a servo motor controlled by the ESP32 microcontroller.[4] After the vehicle enters the parking area, the gate automatically closes.

When the vehicle approaches the exit section, the first exit gate again captures the license plate using the webcam. The system verifies the vehicle number stored during entry and records the exit time.

Using the stored entry time and the current exit time, the system calculates the total parking duration and generates the parking fee automatically. The calculated information is displayed in the server interface running on the laptop.

A QR code containing the vehicle number, entry time, exit time, and calculated payment amount is generated isin python libraries. The user scans the QR code using a mobile QR scanner application to view the payment verification is completed, the second exit gate opens allowing the vehicle to leave the parking area.

To prevent tailgating, the system implements a dual exit gate mechanism. When a vehicle reaches the first exit gate, it opens and allows the vehicle to move to the payment area near the second gate. The first gate closes immediately after the vehicle passes. Only after payment confirmation does the second gate open, ensuring that only one vehicle exists at a time.

## IV. BLOCK DIAGRAM

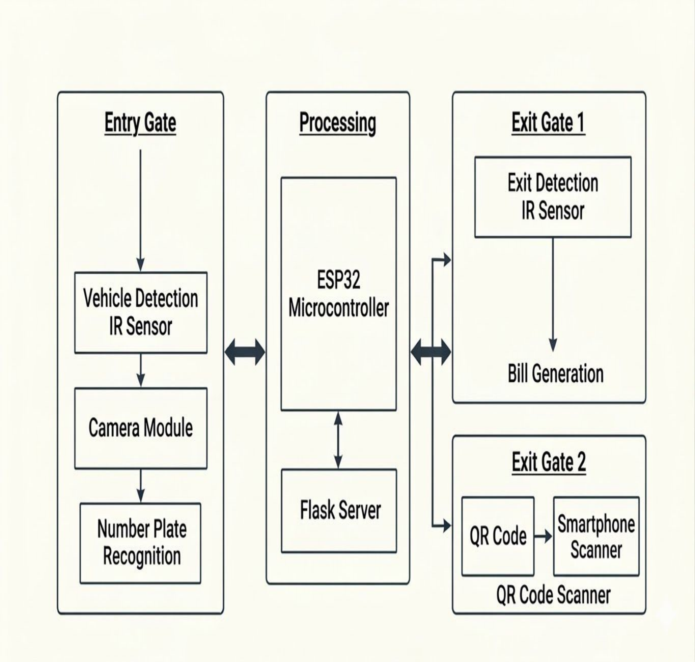

Fig1: Block diagram of Auto-Timed Smart Parking Bill Generation System

## V. HARDWARE COMPONENTS

### 1) *ESP32 MICROCONTROLLER*

The **ESP32 microcontroller** acts as the central control unit of the system. It receives signals from the IR sensors and controls the servo motors to operate the entry and exit gates. The ESP32 is selected due to its high processing capability and compatibility with IoT- based applications.

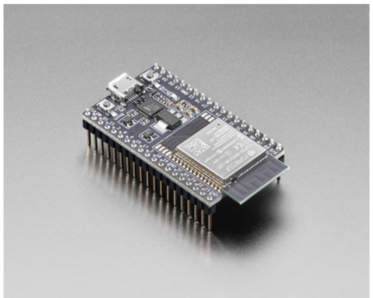

Fig2: ESP32 Micro Controller

### 2) *INFRARED (IR) SENSORS*

The **infrared (IR) sensors** are used to detected the presence of vehicles at the entry and exit gates. When a vehicle interrupts the infrared beam, the sensor sends a signal to the ESP32, triggersthe operation automatically.

Fig3: IR Sensor

### *3) SERVO MOTORS*

The servo motors are responsible for opening and closing the parking gates. In this system, three servo motors are used to control one entry gate and two exit gates. The motors operate based on control signals received from the ESP32 microcontroller.

Fig4: Servo Motors

### *4) WEBCAM*

A **webcam** is used to captured vehicle images at the entry and exit gates. The capture images are processed using Optical Character Recognition (OCR) to detect the vehicle license plate number, which is then used for recording entry and exit times and generating the parking bill.

Fig5: Webcam

## VI. SOFTWARE USED

### *1) Visual Studio Code (VS Code)*

Visual Studio Code (VS Code) is used as the development environment for implementation and executing the python program. It provides a user-friendly platform for writing, debugging, and running the code required for the system operation. The Flask server and other supporting modules are executed through this environment to display and manage the system outputs.

### 2) *Python Programming Language*

The proposed smart parking system is primarily implemented using the python programming language. Python is used to develop the main program that performs vehicle license plate recognition, entry and exit time recording, parking duration calculation, and parking bill generation. The system process images captured by the webcam and applies Optical Character Recognition (OCR) techniques to extract the vehicle license plate number. The recognized number plate is then stored along with the corresponding entry and exit times in the system database for further processing.

### 3) *Flask Web Server*

The Flask web framework is use to create a lightweight server that manages the storage and retrieval of vehicle data. The server records the entry and exit information of vehicles and calculates the total parking duration based on the stored entry and exit times.[5] Using this information, the system generates the corresponding parking charges and manages the communication between different modules of the parking system.

### 4) *QR Code Generation and Scanning*

After the parking bill is calculated, the system generates a QR code containing important parking details such as the vehicle number, entry time, exit time, and total parking fee. This QR code is displayed on the system interface and can be scanned by users using a mobile QR code scanner application to view the payment details and compete the parking payment process.[6]

## VII. RESULT

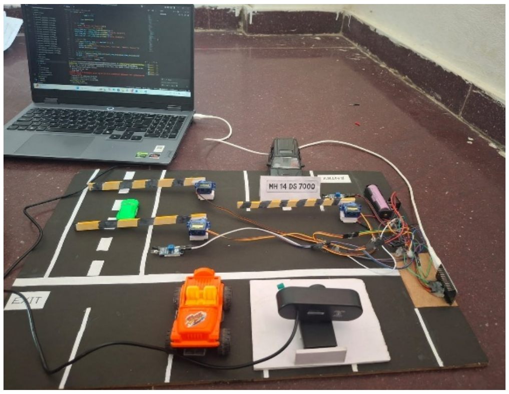

Fig6 :Project Kit

Fig7 : Entry Gate Output

Fig8 : Exit Gate1 Output

Fig9 : Exit Gate2 Output

## VIII. CONCLUSION

This paper presented an auto times smart parking bill generation system based on OCR- based license plate recognition and dual exit gate mechanism for anti-tailgating. The system automates vehicle identification, parking time calculation, billing generation, and payment verification. By integrating ESP32, IR sensors, servo motors, webcam, and a Flask-based backend server, the proposed system provides an efficient parking management solution. The dual exit gate mechanism enhances security by preventing unauthorized vehicle exits through tailgating. The results demonstrate that the system reduces manual effort and improves the efficiency of parking operations. Future work may focus on integrating online payment gateways and improving OCR performance under different environmental conditions.

## REFERENCES

[1] Y. Geng and C. Cassandras, "A New Smart Parking System Infrastructures and Implementation," Procedia- Social andBehavioural Sciences, vol. 54,pp. 1278-1278, 2012.
[2] M. Idris, Y. Leng, E. Tamil, N. Noor, and Z. Razak, "Car Park System: A Review ofSmart Parking System and Its Technology", Information Technology Journal, vol. 8, no.2, pp. 101-113, 2009.
[3] S. Du, M. Ibrahim, M. Shehata, and W. Badawy,"Automatic License Plate Recognition (ALPR): A State-of-the-Art Review,"IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 2, pp. 311–325, 2013.
[4] Espressif Systems,"ESP32 Technical Reference Manual,"Espressif Systems, 2020.
[5] M.Grinberg,Flask Web Development: Developing Web Applications with Python,O'Reilly Media, 2018.
[6] ISO/IEC18004,"QR Code Bar Code Symbology Specification,"International Organization for Standardization, 2015.