Web development based parking system

Submitted by

K.V. RAJEEV REDDY

K. SUJITH REDDY

M.MANJUNATH

Under the Guidance of

Dr. T Nagarjuna

Assistant Professor



Department of Electrical, Electronics and Communication Engineering

GITAM School of Technology

GITAM

(DEEMED TO BE UNIVERSITY)

(Estd. u/s 3 of the UGC act 1956)

NH 207, Nagadenehalli, Doddaballapur taluk, Bengaluru-561203 Karnataka, INDIA.



Department of Electrical, Electronics and Communication Engineering

GITAM School of Technology

DECLARATION

I/We declare that the project work contained in this report is original and it has been done by me under the guidance of my project guide.

Date:

Registration No. Student Name Signature

BU21EECE0100167 K.Rajeev Reddy

BU21EECE0100222 K. Sujith Reddy

BU21EECE0100503 M. Manjunath

Department of Electrical, Electronics and Communication Engineering

GITAM School of Technology, Bengaluru-561203



CERTIFICATE

This is to certify that K Rajeev Reddy bearing BU21EECE0100167, K. Sujith Reddy bearing BU21EECE0100222, M.Manjunath bearing BU21EECE0100503 has satisfactorily completed Mini Project Entitled in partial fulfillment of the requirements as prescribed by University for VIIth semester, Bachelor of Technology in “Electrical, Electronics and Communication Engineering” and submitted this report during the academic year 2024-2025.

[Signature of the Guide] [Signature of HOD]

Table of contents

[**Chapter 1: Introduction 1**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.gjdgxs)

[1.1 Overview of the problem statement 1](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.30j0zll)

[1.2 Objectives and goals 1](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1fob9te)

[**Chapter 2 : Literature Review 2**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.3znysh7)

[**Chapter 3 : Strategic Analysis and Problem Definition 3**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.2et92p0)

[3.1 SWOT Analysis 3](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.tyjcwt)

[3.2 Project Plan - GANTT Chart 3](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1t3h5sf)

[3.3 Refinement of problem statement 3](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.2s8eyo1)

[**Chapter 4 : Methodology 4**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.17dp8vu)

[4.1 Description of the approach 4](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.3rdcrjn)

[4.2 Tools and techniques utilized 4](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.26in1rg)

[4.3 Design considerations 4](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.lnxbz9)

[**Chapter 5 : Implementation 5**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1ksv4uv)

[5.1 Description of how the project was executed 5](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.44sinio)

[5.2 Challenges faced and solutions implemented 5](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.2jxsxqh)

[**Chapter 6:Results 6**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.z337ya)

[6.1 outcomes 6](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.3j2qqm3)

[6.2 Interpretation of results 6](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1y810tw)

[6.3 Comparison with existing literature or technologies 6](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.2xcytpi)

[**Chapter 7: Conclusion 7**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1ci93xb)

[**Chapter 8 : Future Work 8**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.2bn6wsx)

[Here write Suggestions for further research or development Potential improvements or extensions 8](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.qsh70q)

[**References 9**](file:///C:\Users\DELL\AppData\Local\Microsoft\Windows\INetCache\IE\1P0ZG182\4g_report%5b1%5d.docx#_heading=h.1pxezwc)

# Chapter 1: Introduction

# 1.1 Overview of the Problem Statement

# The problem addressed by the project is the insufficient number of parking spaces in urban areas, which has become more critical with the rising number of vehicles. Current parking solutions do not efficiently manage parking spaces, leading to traffic congestion, wasted time, and frustration among drivers. The project aims to mitigate these issues through a web-based automated parking system that leverages real-time monitoring and smart technologies such as IoT, image processing, and GPS. This system provides real-time information on parking availability and integrates an automated payment process, thereby simplifying the parking experience for users.

# The system also offers cost efficiency for parking space owners, providing them with a means to generate revenue by offering a convenient service to users. In addition, the system can operate both online and offline, making it accessible under various conditions.

# .

# 1.2 Objectives and Goals

# A **Web Development-Based Parking System** aims to provide a user-friendly platform for efficient parking space management, allowing real-time tracking, reservations, and secure payments. It also enhances security through AI-based vehicle recognition and offers dynamic pricing for optimal utilization.

# Enhanced User Experience: Simplifying the process of finding, reserving, and paying for parking spots.

# Efficient Parking Space Management: Providing real-time availability updates and reservation options to ensure optimal use of parking spaces.

# Enhanced Security and Enforcement: Utilizing technology to ensure that parking spots are used efficiently and securely.

# Revenue Generation: Allowing parking spot owners to monetize their spaces with minimal investment.

# Additional goals include improving operational efficiency, scalability for various locations (e.g., malls, offices, events), and reducing the overall cost of managing parking spaces​.

# Chapter 2: Literature Review

# An IoT based Intelligent Parking System for the Unutilized Parking Area with Real-Time Monitoring using Mobile and Web Application.

* Rapid Development of Smart Parking System with Cloud-based Platforms
* A Novel Green IoT-Based Pay-As-You-Go Smart Parking System

**Existing Implementations – Products| Opensource| GitHub etc**

<https://github.com/SLoharkar/Online-Parking-System>

<https://github.com/swetabehera/Smart-Parking-System-Website>

# Chapter 3: Strategic Analysis and Problem Definition

# 3.1 SWOT Analysis

# The SWOT analysis identifies:

# Strengths: The system's ability to offer real-time tracking, improve user experience, and provide automated invoicing.

# Weaknesses: Initial setup costs and possible resistance to adoption in some markets.

# Opportunities: Expansion to other locations, integration with other smart city infrastructure, and potential for future upgrades (e.g., AI-driven predictive models for parking demand).

# Threats: Competition from other parking solutions and potential technological failures (e.g., GPS inaccuracies)

# 3.2 Project Plan - GANTT Chart

# The GANTT chart shows key activities across a three-month timeline, breaking down the project into:

# Design Phase: Initial architecture and user interface development.

# Development Phase: Coding and integration of IoT, image processing, and payment systems.

# Testing Phase: Iterative user testing to refine the features and solve bugs.

# Deployment Phase: Final implementation in target environments like malls or office buildings​.

# 3.3 Refinement of Problem Statement

# The problem statement was refined to focus on creating a system that not only addresses the scarcity of parking but also improves traffic flow by reducing the time users spend looking for parking. The system enables users to reserve parking spots in real-time and ensures optimal use of the space by providing real-time updates​.

### Chapter 4: Methodology

#### 4.1 **Description of the Approach**

The development of the web-based parking system follows a structured, **Agile methodology** for iterative and continuous improvement throughout the project lifecycle. This approach involves the following stages:

* **Requirement Analysis**: Understanding the needs of both end-users (customers) and system administrators to define features like real-time parking availability, reservations, dynamic pricing, and AI-based vehicle recognition.
* **System Design**: Dividing the project into key modules: **Frontend**, **Backend**, **Database**, and **AI Integration**. The system architecture follows a **Model-View-Controller (MVC)** structure to ensure modularity and maintainability.
* **Development**: Implementing the frontend for user interaction, the backend for business logic, database integration for persistent data storage, and AI features for vehicle recognition.
* **Testing and Iteration**: Rigorous testing is conducted during each sprint to ensure functionality, security, and performance. The system undergoes unit testing, integration testing, and user acceptance testing (UAT).
* **Deployment**: After successful testing, the system is deployed to cloud services to ensure scalability, accessibility, and real-time data processing.

The approach ensures that the project progresses in a modular fashion, allowing flexibility for new features and improvements based on user feedback and technological advancements.

#### 4.2 **Tools and Techniques Utilized**

The project uses a combination of modern technologies and techniques, which can be broken down as follows:

1. **Frontend Development:**
   * **HTML5 & CSS3**: To build a responsive and visually appealing user interface.
   * **JavaScript**: For handling user interactions and dynamic content updates.
   * **AngularJS**: A popular frontend framework to create a **single-page application (SPA)** for seamless navigation and real-time updates without full page reloads.
   * **Bootstrap**: For ensuring a responsive design that adapts to various screen sizes.
2. **Backend Development:**
   * **Python (Flask/Django)**: Flask or Django is used as the web framework to handle server-side operations, including parking space management, user authentication, and payment processing.
   * **REST APIs**: Implemented to allow frontend-backend communication and third-party integrations (e.g., payment gateways).
   * **Authentication**: Techniques like JWT (JSON Web Tokens) are utilized to ensure secure user login and session management.
3. **Database Management:**
   * **Firebase/SQL**: Firebase for real-time data management and SQL databases like MySQL/PostgreSQL for structured data storage. Firebase offers live data sync, while SQL databases are used for transactional records such as parking reservations and user profiles.
   * **NoSQL**: Used to store unstructured or semi-structured data such as logs, user behavior patterns, and vehicle images.
4. **AI & Image Processing:**
   * **OpenCV**: For **License Plate Recognition (LPR)**, using computer vision techniques to detect and extract license plate information from vehicle images.
   * **TensorFlow/Keras**: For AI-based vehicle recognition models that improve the system’s accuracy in identifying vehicles.
   * **YOLO (You Only Look Once)**: A pre-trained object detection model to ensure fast and real-time vehicle detection in parking areas.
5. **Payment Gateway Integration:**
   * **Stripe/PayPal**: APIs from payment services such as Stripe or PayPal are integrated to handle online payment transactions securely. SSL encryption and secure payment protocols are followed to ensure user financial data protection.
6. **Cloud Deployment:**
   * **AWS/GCP**: The system is deployed on cloud platforms like **Amazon Web Services (AWS)** or **Google Cloud Platform (GCP)** for scalability and real-time data access.
   * **Load Balancing**: Implemented to manage the traffic load and distribute resources effectively during peak usage.

#### 4.3 **Design Considerations**

Several design considerations were factored in to ensure the system is scalable, secure, user-friendly, and adaptable for future growth:

1. **Scalability:**
   * The system is designed to handle a growing number of users and parking spaces by leveraging cloud infrastructure and **database sharding**. It can easily be scaled up to include additional parking lots, zones, or even cities.
   * **Modular Design**: New features like electric vehicle charging stations or dynamic pricing models can be added without re-engineering the entire system.
2. **Security:**
   * **Data Encryption**: All sensitive data, including user credentials, payment information, and vehicle details, are encrypted using **SSL/TLS** protocols to prevent unauthorized access.
   * **License Plate Recognition**: The system uses LPR to enhance security by automatically granting or denying access to vehicles based on pre-defined user roles and vehicle registration.
   * **Two-Factor Authentication (2FA)**: Optionally available for user accounts to ensure an additional layer of security for users.

 Performance Optimization:

* Caching: Techniques like HTTP caching and in-memory databases (Redis) are used to reduce server load and ensure fast responses to user queries.
* Load Testing: The system undergoes stress testing to ensure it can handle high traffic without degradation in performance, especially during peak hours.
* Green Parking Initiatives:
* Special parking spots are allocated for electric vehicles, and the system can provide insights into energy-efficient parking choices, like parking close to EV charging stations or in areas designed to reduce vehicle idling time.

### Chapter 5: Implementation

#### 5.1 **Description of How the Project is Being Executed**

The project is currently in the development phase, following a structured plan from design to eventual deployment. After finalizing the system architecture, the team has begun integrating key technologies such as IoT devices for parking space monitoring, image processing for vehicle recognition, and web-based interfaces. Several rounds of iterative testing are being conducted to ensure that all components function seamlessly. The project incorporates continuous feedback loops to detect and resolve bottlenecks, such as slow data synchronization and GPS inaccuracies, which are addressed in each iteration.

#### 5.2 **Challenges Faced and Solutions Being Implemented**

Some ongoing challenges include:

* **System Latency**: The team has encountered delays in updating parking availability in real-time, caused by server-side processing issues.

**Solution**: Firebase is being used to optimize backend processes for faster synchronization, and efforts are underway to reduce latency by implementing more efficient data caching mechanisms.

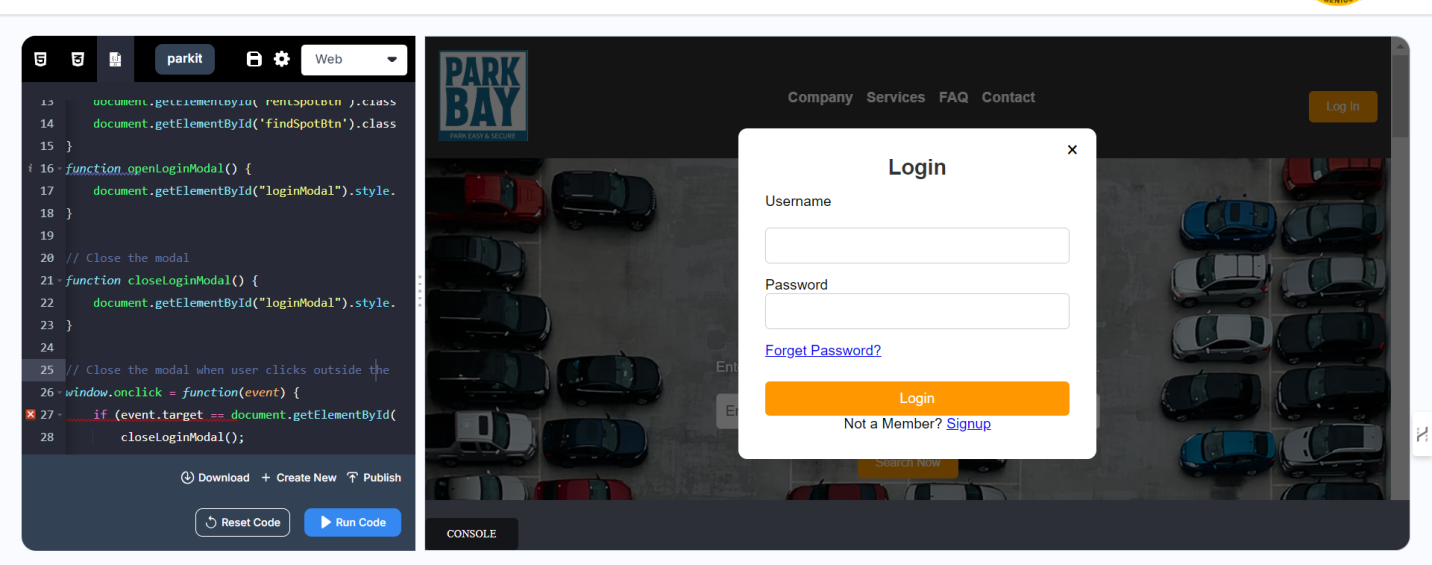
* **Integration of Multiple Technologies**: Ensuring that IoT, GPS, and image processing technologies work in harmony has proven to be complex.

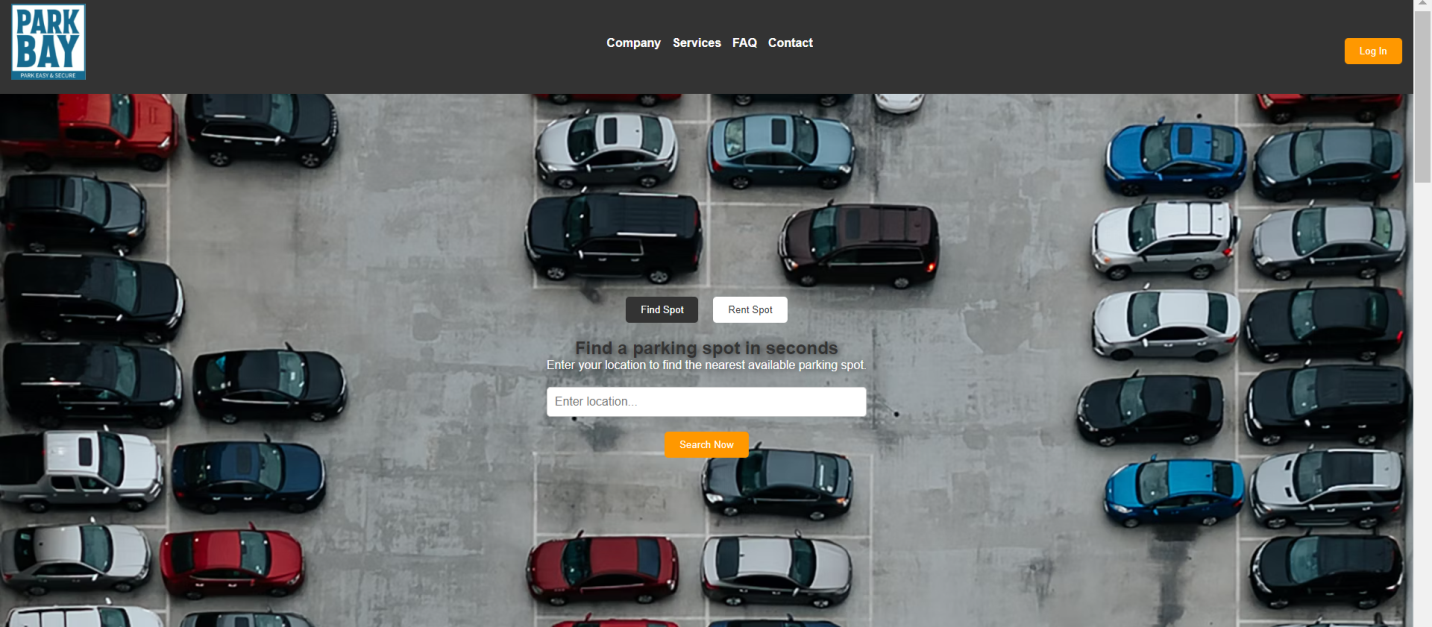
**Solution**: The project employs a modular integration approach, refining APIs to isolate and address issues in individual components, and conducting iterative testing to ensure smooth operation.

### Chapter 6: Results

#### 6.1 **Expected Outcomes**

Upon completion, the project aims to deliver a fully functional web-based parking system capable of providing real-time updates on parking availability, automating billing, and offering location-based services. The system is expected to improve user experience, streamline parking space management, and offer a cost-effective solution for parking lot owners



.

#### 6.2 **Interpretation of Anticipated Results**

The integration of real-time monitoring, IoT, and user-friendly web interfaces is expected to significantly reduce the time users spend searching for parking spots, while also optimizing traffic flow in busy areas. The system’s potential adaptability for different environments, such as shopping malls, office complexes, or city centers, suggests it could be a versatile and scalable solution.

#### 6.3 **Comparison with Existing Literature or Technologies**

Compared to existing parking management solutions, this system distinguishes itself by incorporating IoT devices for real-time monitoring and using image processing for vehicle identification. Unlike many existing systems, this project also aims to include offline functionality to maintain usability during connectivity issues, making it accessible to smaller parking operators. Its cost-effectiveness adds to its appeal for a wider range of users.

### Chapter 7: Conclusion

The project is expected to conclude that automated, web-based parking systems provide considerable advantages in reducing parking congestion, improving user satisfaction, and increasing revenue for parking lot owners. The system is being designed with scalability in mind, allowing it to be adapted for various urban settings. It holds promise for supporting smart city initiatives through its real-time data capabilities and modularity.

### Chapter 8: Future Work (In Development Phase)

In the future, the parking system could include AI to adjust prices based on demand. This means prices could change automatically depending on how busy the parking lot is, helping lot owners earn more and encouraging users to park in less crowded areas.

The system could also use blockchain technology to make payments safer and more transparent. This would allow for secure, fast transactions and better protection of user data.

Another improvement could be using augmented reality (AR) to guide users to parking spots. Users could get real-time visual directions through their phones, making it easier to find available spots or locate their parked car.

As self-driving cars become more common, the system could adapt to let these vehicles park themselves. Users could drop off their cars, and the system would handle the parking, making it more efficient for autonomous vehicles.

Lastly, the system could focus on eco-friendly features like encouraging users to park in ways that reduce fuel use and even using solar power for energy in parking lots. This would help make the system more sustainable and environmentally friendly.