System and method for resolving the parking issues using smart vehicle parking

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE AWARD OF DEGREE OF

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE

Submitted by

ISHITA SRIVASTAVA (2100290120088)

KRITIKA (2100290120098)

GURPREET KAUR (2100290120082)

Supervised by

DR. RISHABH

Associate Professor

Session 2024-25

DEPARTMENT OF COMPUTER SCIENCE

KIET GROUP OF INSTITUTIONS, GHAZIABAD

(Affiliated to Dr. A. P. J. Abdul Kalam Technical University, Lucknow, U.P., India)

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Signature:-
Name:- Ishita Srivastava
Roll No:- 2100290120088

Signature:-
Name:- Gurpreet Kaur
Roll No:- 2100290120082

Signature:-
Name:- Kritika
Roll No:- 2100290120098

Date:- 10/05/2025

CERTIFICATE

This is to certify that Project Report entitled " **System and method for resolving the parking issues using smart vehicle parking** " which is submitted by Ishita Srivastava, Kritika, Gurpreet Kaur in partial fulfillment of the requirement for the award of degree B. Tech. in Department of Computer Science of Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my supervision. The matter embodied in this report is original and has not been submitted for the award of any other degree.

Date:- 10/05/2025

Supervisor:

Signature:- 10/05/2025

Name:- Dr. Rishabh

Designation:- Associate Professor

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B.Tech Project undertaken during B. Tech. Final Year. We owe a special debt of gratitude to Professor Rishabh, Department of Computer Science, KIET, Ghaziabad, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen the light of day.

We also take the opportunity to acknowledge the contribution of Dr. Ajay Kr. Shrivastava, Head of the Department of Computer Science, KIET, Ghaziabad, for his full support and assistance during the development of the project. We also do not like to miss the opportunity to acknowledge the contribution of all the faculty members of the department for their kind assistance and cooperation during the development of our project.

Last but not least, we acknowledge our friends for their contribution to the completion of the project.

Signature:- Ishita
Name:- Ishita Srivastava
Roll No:- 2100290120088

Signature:- Gurpreet
Name:- Gurpreet Kaur
Roll No:- 2100290120082

Signature:- Kritika
Name:- Kritika
Roll No:- 2100290120098

Date:- 10/05/2025

ABSTRACT

Traffic congestion, illegal parking, and safety problems have resulted from the imbalance between the increasing number of vehicles and the restricted number of parking spots in metropolitan areas. Conventional parking infrastructure is frequently ineffective, resulting in higher fuel use, wasted time, and financial losses. By offering an intelligent, Internet of Things-enabled parking solution that maximizes parking space use without necessitating extra infrastructure development, the initiative seeks to address these issues.

The project links up automobile owners looking for parking spots with owners of open spots who are prepared to lease their spots. To provide a flawless parking experience, the system combines Django (backend), Firebase (real-time data sync), and IoT sensors (parking space detection). Real-time parking availability updates are provided by ultrasonic and infrared sensors, cutting down on needless fuel use and cruise time.

ParkEzy encourages sustainable urban mobility through contactless purchases, automated booking, and geolocation-based slot discovery. By increasing transportation efficiency and lowering environmental impact, this project supports the Sustainable Development Goals (SDGs 11 and 12: Sustainable Cities, Responsible Consumption, and Industry & Innovation, respectively).

The implementation of the project is expected to reduce urban traffic congestion, enhance parking safety, and provide additional income opportunities for parking space owners, contributing to a smarter and more sustainable urban future.

TABLE OF CONTENTS

Page No.

LIST OF FIGURES

LIST OF TABLES

SDG MAPPING WITH JUSTIFICATION

SDG 11: Sustainable Cities and Communities

1. **Lessening Traffic Congestion:** The project makes the most of the current infrastructure by letting customers rent parking spots, which lessens needless traffic and movement.

2. **Improving Urban Mobility:** Sustainable urban planning is supported by efficient parking management, which guarantees a smooth transit flow.

3. **Reducing Accidents & Illegal Parking:** Well-organized parking lots reduce careless parking, which makes the roads safer and lowers the number of accidents.

4. **Encouraging Smart Cities:** IoT integration advances the goal of smart urban infrastructure, which makes cities more effective and sustainable.

SDG 12: Responsible Consumption and Production

1. **Optimizing Space Utilization:** This promotes the effective use of parking spots that are available, which lessens the demand for additional parking lots and conserves land.

2. **Reducing Fuel Consumption & Emissions:** By spending less time looking for parking, drivers reduce their carbon footprint and fuel waste.

3. **Promoting Community Participation:** This allows people to make money off of the locations they don't use, which encourages the use of shared resources.

4. **Digitalization for Paperless Transactions:** By using Django and IoT for online administration, the project lessens the need for paper records and physical tickets.

SDG 9: Industry, Innovation, and Infrastructure

1. **IoT Integration for Smart Parking:** ParkEzy automates parking and increases efficiency by using ESP8266, IR sensors, and ultrasonic sensors.

2. **Real-Time Data Processing:** Firebase provides real-time parking status updates, which improve user and municipal planner decision-making.

3. **Promoting Technological Development:** Innovative urban infrastructure is encouraged by the MQTT protocol's use for data transfer.

4. **Scalability for Future Expansion:** To promote long-term sustainability, the model can be extended to incorporate AI-driven predictive parking and EV charging stations.

CHAPTER 1

INTRODUCTION

1.1 Background

There is now a major disparity between the supply of parking spots and the growing demand due to the quick rise in the number of vehicles, which causes traffic jams, obstructions, and chaotic parking situations. In congested marketplaces, tourist attractions, and urban areas, the problem is exacerbated by the absence of structured parking structures, which leads to chaotic on-street parking, higher fuel usage, and needless emissions.

The predicament for car owners is made worse by traffic, the possibility of getting a challan (fine), and safety issues. Finding appropriate parking spaces is made more difficult for vehicles by the lack of regulated parking slot dimensions. By making effective use of currently underutilized spots, ParkEzy aims to solve these parking issues without requiring further infrastructure investment.

1.2 Introduction

The project **is a** smart parking management system **designed to tackle parking saturation by** consolidating underutilized private and commercial spaces into a structured parking network**. The platform enables** vehicle owners to find and reserve parking spots in real-time**, while allowing** landowners to monetize their vacant spaces**. This solution aligns with the** Smart Cities initiative and Sustainable Development Goals (SDGs) **by optimizing existing resources rather than creating additional infrastructure**.

To make sure everything feels safe, reliable, and trustworthy, ParkEzy has a solid rating, review, and verification system in place. It also uses smart sensors, live data, and an easy-to-use interface to make the whole parking experience smoother and more convenient for everyone.

1.3 Project Category

i. Smart Parking & Urban Mobility - This initiative uses technology to maximize parking space use.

ii. IoT & Automation: For real - time parking monitoring, ESP8266, IR sensors, and ultrasonic sensors are used.

iii. Web development - Django was used to create an effective parking space booking and administration system.

iv. Sustainable Solution - It focuses on SDGs 9 (Innovation), 11 (Sustainable Cities), and 12 (Responsible Consumption) are all supported by the Smart City Initiative.

1.4 Objectives

1. Maximize Parking Space Utilization: To lessen parking shortages, make use of underutilized private and commercial spaces.

2. Reduce Traffic Congestion: Smoother traffic flow results from fewer cars looking for spots and parking on the street.

3. Improve Security & Reliability: To make parking safer, put in place a system of reviews, ratings, and verification.

4. Offer Real-Time Parking Solutions: Utilize a web-based platform to help users locate, reserve, and oversee parking spaces.

5. Use IoT for Smart Parking: For precise space automation and monitoring, use infrared and ultrasonic sensors.

6. Encourage Sustainable Urban Mobility: By reducing needless car travel, you may cut down on pollution and fuel waste.

7. Provide Extra Revenue Streams: It also gives landowners a chance to earn some extra money by renting out their empty parking spots instead of letting them sit unused.

8. SDG Objectives: It focuses on these 3 SDGs - **SDGs 9 (Innovation), 11 (Sustainable Cities), and 12 (Responsible Consumption).**

CHAPTER 2
LITERATURE REVIEW

2.1 Introduction

India, as the second most populated country globally, grapples with accommodating over 40 million vehicles amid urbanization and economic growth. This surge in vehicle ownership, even during economic downturns, has led to severe parking issues across all urban areas. Parking problems manifest in two forms: off-street parking, common in shopping malls and large offices, and on-street parking, dictated by market forces, often leading to congestion. Overlooking parking in urban policy frameworks results in issues such as poor transport quality, land misuse, and environmental costs. To address these challenges, expanding parking spaces, enhancing accessibility, and optimizing existing infrastructure are essential. Additionally, incentivizing public transportation use, implementing variable parking pricing, and strategically pricing parking in commercial areas can effectively manage demand.

2.2 Review

The research on parking issues in India delves into a multifaceted problem exacerbated by rapid urbanization, population growth, and increasing vehicle ownership. This comprehensive review encapsulates key findings and solutions proposed across several papers, highlighting the urgency of addressing parking scarcity and its implications on urban mobility. India's urban landscape transformation is coupled with unprecedented population and vehicular traffic growth, straining existing parking infrastructure. In response, innovative technologies have been explored to optimize parking utilization. The paper by Manjaly and Joseph **emphasizes** the urgency of the problem, with annual monetary losses due to parking issues calculated as 1.56% of India's GDP. Their proposed smart parking systems offer promising solutions to allocate parking spaces efficiently, enhancing convenience and reducing traffic congestion. Mobile applications emerge as valuable tools for parking management, as discussed in the study by Parmar et al.), allowing users to locate nearby parking facilities, reserve spots, and receive navigation guidance. Case

studies demonstrate the efficacy of such applications in enhancing the overall parking experience and maximizing the utilization of existing parking infrastructure. Aligning policy frameworks with urban planning objectives, as advocated by Peñalosa, is crucial. Municipal authorities play a pivotal role in formulating regulations and guidelines to govern parking supply, pricing, and enforcement effectively. Strategies such as parking supply mandates for new developments, and dynamic pricing. Integrated transportation planning prioritizing sustainable modes of mobility, including public transit, walking, and cycling, is emphasized. The lack of parking spaces for road freight transport, discussed by Lizbetin and Bartuska", underscores the broader challenges. Promoting alternative transportation options and establishing transit-oriented development principles can reduce dependency on private vehicles and alleviate parking demand.

Surveys and empirical studies provide insights into public perceptions and behavior regarding parking. Understanding user preferences and travel patterns, as mentioned in Ahmed's examination of the car parking problem in urban areas, is essential for designing tailored interventions. Environmental sustainability, highlighted in various studies, is another critical dimension of parking management, with measures such as green parking infrastructure and electric vehicle charging stations contributing to climate mitigation efforts.

Public-private partnerships (PPPs) in financing and implementing parking infrastructure projects, as proposed by Ahmedi", are advocated. Collaborative models involving government agencies, private developers, and technology providers enable resource mobilization for scaling up parking solutions. Community engagement and stakeholder participation are integral, fostering consensus-building and inclusive decision-making processes.

2.3 Summary

In summary, the parking challenges in highly populated Indian cities are multifaceted and require a comprehensive solution. The saturation of parking spaces, lack of regulation in parking tariffs, and environmental concerns have underscored the need for innovative approaches to parking management. Efforts to improve parking must be accompanied by effective policy measures, public awareness campaigns, and technological innovations. By addressing parking issues comprehensively, India can mitigate traffic congestion, reduce environmental impact, and enhance the overall quality of urban life.

Efforts to improve parking management must be accompanied by effective policy measures, public awareness campaigns, and technological innovations. By addressing parking issues comprehensively, India can mitigate traffic congestion, reduce environmental impact, and enhance the overall quality of urban life. Integrating insights from studies by Manjaly and Joseph", Parmar et al., Peñalosa", Lizbetin and Bartuska", and Ahmed", this review provides a roadmap for policymakers, urban planners, and stakeholders seeking to create more accessible, efficient, and livable cities in India and beyond.

2.4 Conclusion

In conclusion, the proposed invention addresses the pressing issues associated ever-expanding number of vehicles and the limited availability of parking spaces. areas grapple with the challenges of congestion and insufficient parking facilit innovation aims to streamline the process by aggregating vacant areas under a platform dedicated to parking solutions, The current imbalance between the increasing number of vehicles and the stagnant park infrastructure results in chaotic situations during peak hours, causing roadblocks a inconvenience. Our approach involves consolidating individual entities' vacant spac under a common label or umbrella, making them easily accessible for ing. This no only resolves the problem of saturated parking spaces but also aligns with the vision of Smart City initiatives and Sustainable Development goals. Moreover, landowners have the opportunity to generate additional revenue by participating in this shared parking system.

CHAPTER 3
PROPOSED SYSTEM

3.1 Proposed System

The project proposes an innovative parking management system that leverages IoT technology and a user-friendly web platform to address urban parking challenges effectively. ParkEzy is made to help people find parking more easily and make use of empty spots that often go unused. It's simple—sensors are placed in parking areas to check if a spot is free or taken. This information is sent using a small device called ESP8266, so the app always shows live updates.

We built a website using Django where users can search for parking, book a spot, and even pay online. It saves time and avoids the stress of driving around looking for parking.

There's also a rating and review feature where people can share their experience, which helps others choose safe and good spots. Plus, if someone has an empty parking space, they can rent it out and earn a little extra money.

Overall, ParkEzy helps reduce traffic, saves fuel, and supports a cleaner, smarter way to move around the city.

3.2 Unique Features

1. **Real-Time Parking Monitoring:** The project utilizes IoT technology with ultrasonic and infrared sensors installed in parking spaces to provide real-time data on parking availability. Users can instantly access this information through the web application, allowing them to quickly locate and reserve available spots, which significantly reduces the time spent searching for parking and minimizes traffic congestion**.**
2. **User Rating and Review System**: To enhance safety and build trust within the community, the platform incorporates a robust rating and review system. Users can leave feedback about their parking experiences, helping others make informed decisions. This

feature fosters a sense of accountability among space providers and promotes high-quality service.

3. **Seamless Integration of Payment Solutions:** The platform supports multiple payment options, including digital wallets, credit/debit cards, and mobile payment solutions. This flexibility simplifies the booking process, allowing users to choose their preferred method of payment and promoting a cashless transaction environment.
4. **User-Friendly Interface**: Designed with a clean and intuitive interface, the ParkEzy web application enhances user experience by making navigation straightforward. Users can easily search for parking spots, make reservations, and manage their accounts, catering to users of all technological backgrounds.
5. **Sustainable Urban Mobility:** The project promotes sustainable urban mobility by maximizing the efficient use of existing parking resources without necessitating new infrastructure development. By connecting vehicle owners with underutilized spaces, it contributes to reduced traffic congestion, decreased emissions, and overall environmental sustainability.
6. **Integration with Smart City Initiatives:** The system aligns with smart city frameworks, enabling local governments to leverage ParkEzy's data for urban planning and traffic management. This collaboration can lead to better resource allocation and improved urban infrastructure over time.
7. **Enhanced Security Features:** ParkEzy implements security measures, such as user verification and GPS tracking, to ensure the safety of parked vehicles. This added layer of security instills confidence in users, knowing their vehicles are protected while parked.
8. **Notifications**: Users will get notifications and will receive alerts about parking availability, booking confirmations, and payment reminders. This feature keeps users informed and engaged with the platform, enhancing their overall experience.
9. **Community Engagement and Support**: ParkEzy fosters a sense of community by encouraging users to participate in forums and discussions. Users can share tips, report issues, and engage with each other, creating a supportive environment for all participants.

CHAPTER 4

REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

4.1 Feasibility Study (Technical, Economical, Operational)

The feasibility study evaluates the viability of the project based on technical, economic, and operational aspects to ensure its successful implementation and sustainability.

1. **Technical Feasibility :-** It leverages IoT, cloud computing, and web technologies to create an efficient parking management system.

- Hardware Integration: The system uses ultrasonic sensors, IR sensors, and ESP8266 microcontrollers to detect real-time parking space availability.
- Software & Database: The platform is built using Django (Python framework) and Firebase for real-time data updates.
- Scalability: The system is designed to handle multiple users and transactions simultaneously, making it scalable for expanding into different cities.
- Security & Reliability: User authentication, encrypted transactions, and a review system ensure data security and reliability.

2. **Economical Feasibility:-**The project is economically viable due to its low implementation cost and revenue-generating model.

- Initial Investment: The primary costs involve sensor deployment, cloud hosting, and web platform development, which are affordable compared to traditional parking infrastructure expansion.
- Revenue Model:
    - Parking Space Owners earn revenue by renting out underutilized parking spaces.
    - Users pay a nominal fee for booking parking spots, generating income for the platform.
    - Premium Features (such as reserved slots, extended time, and special parking areas) provide additional revenue streams.

- Return on Investment (ROI): The system has high revenue potential, making it self-sustainable in the long run.
    - Since the project optimizes existing parking spaces, it eliminates the need for costly infrastructure expansion, making it an economically feasible solution.

3. **Operational Feasibilty:-** Operational feasibility assesses the ease of implementation, management, and user adoption of the system.

- User-Friendly Platform: The web-based system provides an intuitive interface for users to book and manage parking reservations easily.
- Minimal Training Requirement: Vehicle owners and space providers require minimal training, as the system is designed to be simple and automated.
- Automated Monitoring & Alerts: The system automatically updates parking availability and sends notifications to users, reducing manual intervention.
- Security & Trust Mechanism: The rating and review system ensures reliability, encouraging more users to adopt the platform.
- Collaboration with Local Authorities: The project can be integrated with smart city initiatives and municipal partnerships for better implementation and legal compliance

4.2 Software Requirement Specification

This section defines the technical and functional requirements necessary to implement the system.

4.2.1 Data Requirement:-

A. VehicleOwner (Users Seeking Parking)

These are users searching for parking spaces.

Attributes:

- User ID (Primary Key) – Unique identifier for each user
- Name – Full name of the vehicle owner
- Contact Details – Phone number and/or email
- Vehicle Details – Type (sedan, SUV, electric vehicle, etc.)

- Preferences – Covered parking, EV charging, proximity filters, etc.

B. SlotOwner (Users Renting Out Parking Spaces)

These are users providing parking spaces.

Attributes:

- **Slot Owner ID (Primary Key)** – Unique identifier for each owner
- **Name** – Full name of the slot owner
- **Space ID –** Unique identifier for each parking slot
- **Location** – Address or latitude/longitude for geolocation mapping
- **Availability Schedule** – Time slots when the space is free
- **Rental Price** – Hourly or fixed rental charges
- **Payment Details** – Bank or UPI details for receiving payments

C. Booking (Relationship Between VehicleOwner & SlotOwner)

Represents an agreement between a VehicleOwner and a SlotOwner for a parking slot.

Attributes:

- **Booking ID (Primary Key) –** Unique identifier for each booking
- **VehicleOwner ID (Foreign Key)** – Links to the user booking the space
- **SlotOwner ID (Foreign Key)** – Links to the owner of the space
- **Space ID (Foreign Key)** – Links to the specific parking slot
- **Booking Date & Time** – Date and time of booking
- **Duration –** Start and end times of the booking
- **Payment Status –** Pending or Paid
- **Transaction ID –** Unique identifier for payment processing

D. Payment (Transaction Data for Bookings)

Stores details of payments made by users.

Attributes:

- **Transaction ID (Primary Key) –** Unique identifier for each payment
- **Booking ID (Foreign Key)** – Links payment to a specific booking

- **User ID (Foreign Key)** – Links payment to a specific user
- **Amount** – Total amount paid for the parking slot
- **Payment Method** – Credit/Debit card, UPI, Wallet, etc.
- Payment Status – Successful, Failed, or Pending
- Timestamp – Date and time of transaction

E. Geolocation & Navigation Data

Used to display available parking slots on a map and navigate users to the location.

Attributes:

- **Space ID (Foreign Key**) – Links to a parking space
- **Latitude & Longitude** – Stores precise GPS coordinates

4.2.2 Functional Requirement:-

1. User Authentication and Registration:- Users should be able to create accounts, log in, and manage their profiles. Implement secure authentication mechanisms (e.g., OAuth, JWT).

2. Parking Spot Search and Navigation:- Users can search for available parking spots based on location, time, and other filters. Integration with maps (Google Maps, OpenStreetMap) for navigation to the selected spot.

3. Real-Time Availability:- Updates Display real-time availability status for each parking spot. Update availability dynamically as spots are booked or become free.

4. Booking and Payment:- Users can book parking spots in advance. Implement secure payment gateways for booking and payment processing.

5. Notifications:- Send notifications to users for booking confirmation, reminders, and updates. Notify users of any changes in availability or booking status.

6. Rating and Reviews:- Allow users to rate and review parking spots. Aggregate ratings to help users make informed decisions.

7. Admin Dashboard:- Admins can manage parking spaces, view bookings, and handle disputes. Monitor system health and performance.

8. IoT Integration:- If using IoT sensors, integrate them to detect spot availability. Update availability status automatically based on sensor data.

9. Security and Privacy:- Encrypt user data and ensure secure communication. Implement role-based access control (user, admin).

4.2.3 Performance Requirement:-

1. Real-time Parking Availability Updates

- The system must process and update parking slot availability within milliseconds using ESP8266, IR sensors, and Firebase.
- Minimal latency in fetching and displaying data on the web interface.

2. Scalability

- The system should handle multiple users searching for parking spaces simultaneously without performance degradation.
- PostgreSQL must efficiently manage large volumes of parking and transaction data.

3. Optimized Database Queries

- Efficient use of indexes and caching mechanisms to reduce query response time.
- Avoid redundant data fetches with proper ORM (Django ORM) optimizations.

4. Low Power Consumption for IoT Devices

- ESP8266 should be optimized to operate with minimal power consumption while maintaining reliable Wi-Fi communication

4.2.4 Maintainability Requirement:-

1. Modular Code Structure

- Follow MVC architecture in Django to separate business logic, data management, and UI components.

2. Extensibility

- The system should allow future enhancements like dynamic pricing, AI-based parking predictions, and mobile app integration.

3. Automated Testing

- Implement unit testing using Django's built-in test framework to ensure smooth functionality with each update.

4. Code Documentation

- Clear documentation for API endpoints, database models, and IoT communication protocols to ease future modifications.

4.2.5 Security Requirement:-

1. User Authentication & Authorization

- OAuth or JWT authentication for secure login.
- Role-based access control (RBAC) for vehicle owners and space providers.

2. Data Encryption

- Encrypt sensitive user data (passwords, transactions) using AES or SHA-256 hashing.

3. IoT Device Security

- Secure ESP8266 MQTT communication to prevent man-in-the-middle attacks.
- Ensure firmware updates are authenticated to prevent unauthorized modifications.

4. SQL Injection & XSS Prevention

- Use Django's built-in CSRF protection and parameterized queries to prevent attacks.

4.3 SDLC Model Used:- Agile Model

Architecture

- Frontend: HTML, CSS, JavaScript (with Dark Mode UI).
- Backend: Django (Python) with Django REST Framework.
- Database: PostgreSQL (structured parking data).

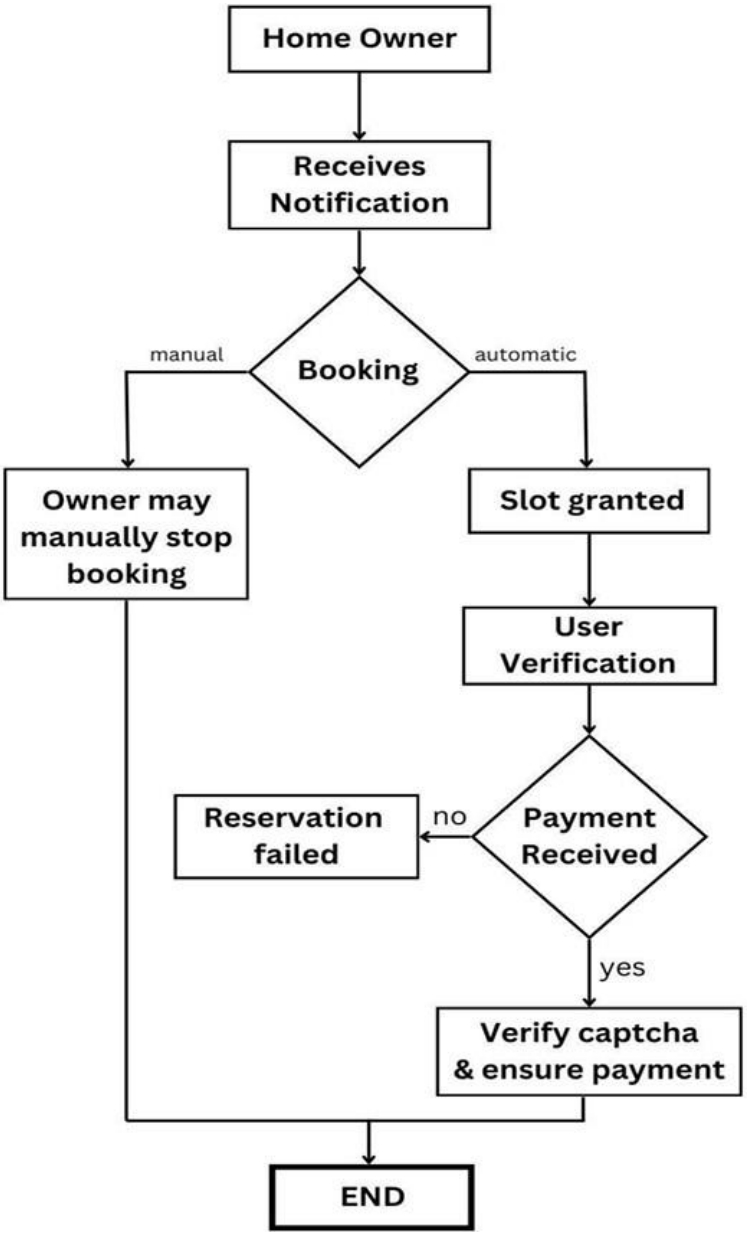
IoT Layer:

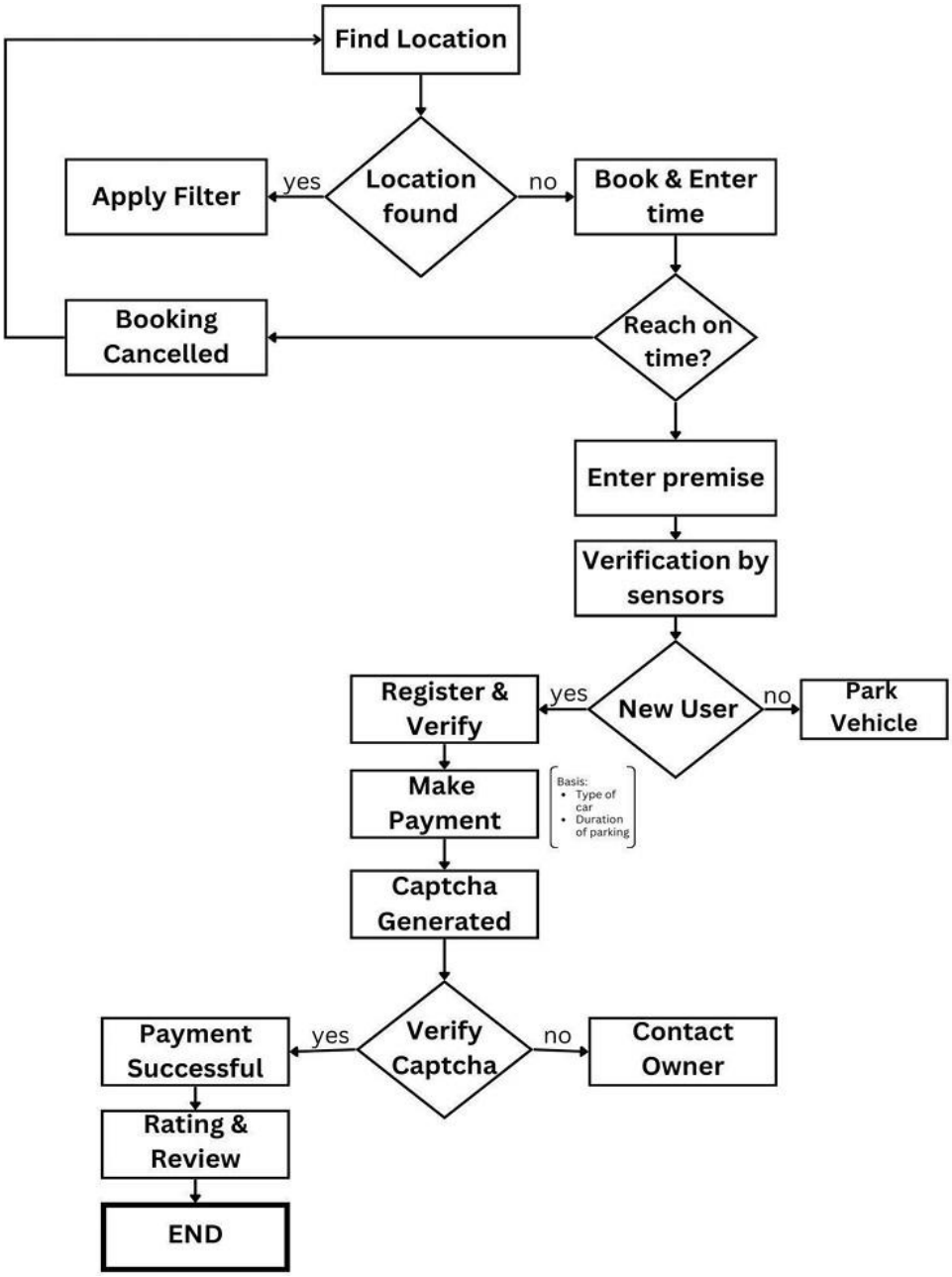
- ESP8266 (Wi-Fi module) + IR & Ultrasonic Sensors for detecting parked cars.
- MQTT Protocol for transmitting real-time sensor data to Firebase.

Workflow

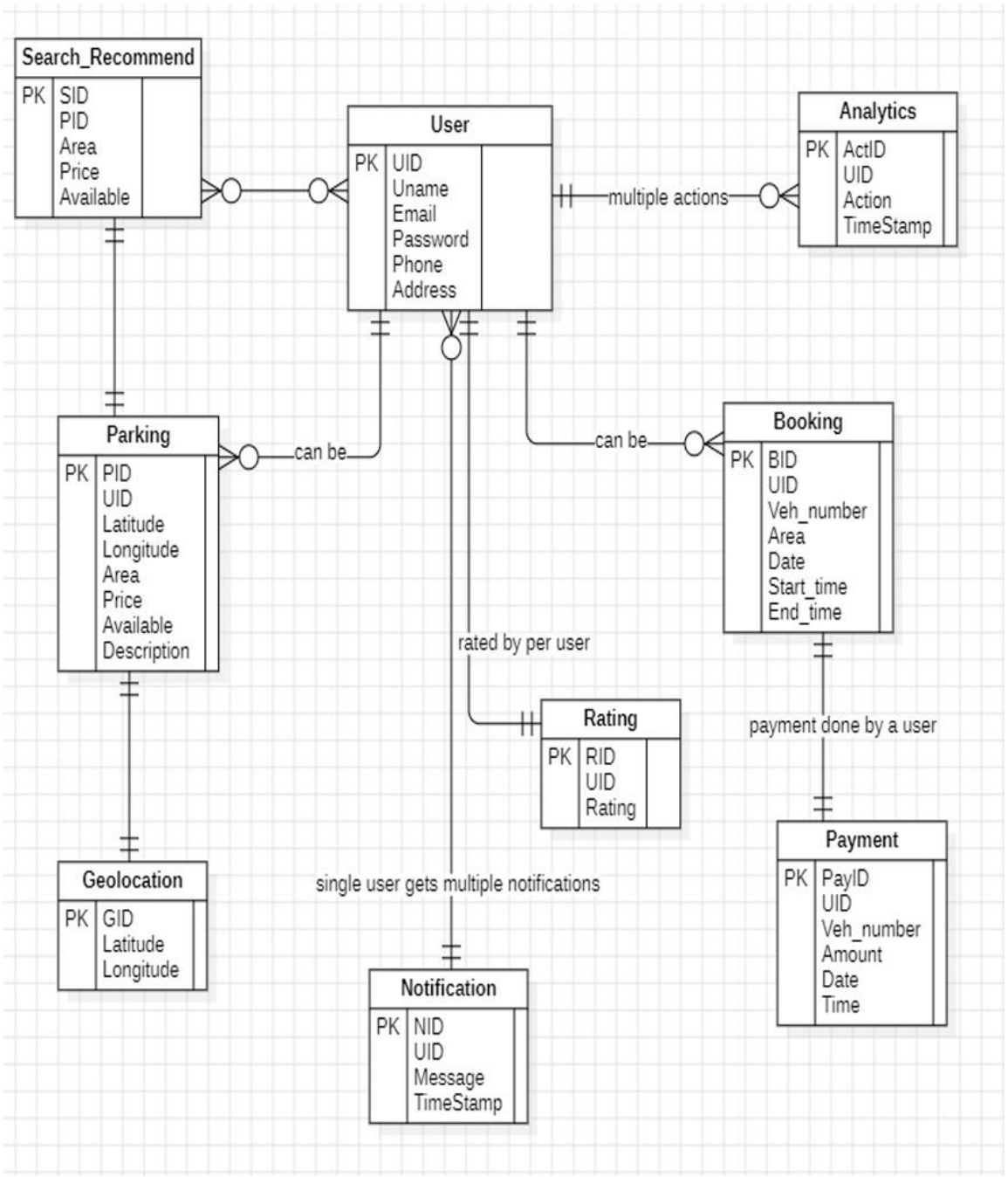
- User searches for an available parking slot → Query sent to the server.
- IoT sensors detect parking occupancy → Data sent to Firebase → Updated in PostgreSQL.

Data Flow Diagram:-




Find Location
Location found
yes
Apply Filter
no
Book & Enter time
Reach on time?
Booking Cancelled
Enter premise
Verification by sensors
New User
yes
Register & Verify
no
Park Vehicle
Make Payment
Basis:
• Type of car
• Duration of parking
Captcha Generated
Verify Captcha
yes
Payment Successful
no
Contact Owner
Rating & Review
END

4.4 Database Design:-



CHAPTER 5

IMPLEMENTATION

5.1 Tools and Technologies Used in ParkEzy Project

The project utilizes a combination of modern tools and technologies to ensure efficient parking management, real-time availability updates, and secure transactions. The key technologies used in this project are categorized as follows:

1. Backend Technologies

- Django REST Framework for API endpoints
- PostgreSQL database for data persistence
- JWT for authentication
- Django Channels for real-time updates (IoT integration)

2. Frontend Technologies

- React.js with hooks for state management
- React Router for navigation
- Axios for API requests
- Dark theme UI matching the provided mockups

3. IoT Technologies (For Smart Parking)

- **ESP8266 (Wi-Fi Microcontroller):**
  - Connects IoT sensors with cloud storage.
  - Sends parking slot availability data to Firebase.

- **Ultrasonic Sensors:**
  - Detects the presence of vehicles in a parking slot.
  - Helps in real-time parking status updates.
- **IR Sensors:**
  - Used for object detection to check slot occupancy.

4. Payment & Authentication Technologies

- **Django-PayPal:**
  - Facilitates secure online payments for parking bookings.
- **JWT (JSON Web Tokens) Authentication:**
  - Ensures secure user login and session management.

5. Development & Collaboration Tools

- **Git & GitHub:**
  - Version control for tracking project updates and collaboration.
  - GitHub repositories store source code and manage contributions.
- **VS Code:**
  - IDEs are used for writing, debugging, and managing the project code efficiently.

CHAPTER 6

TESTING AND MAINTENANCE

6.1 Testing Techniques and Test Cases Used

1 Introduction

A. Test Strategies

a. Functional Testing:

- Ensures that each feature of the application works according to the requirements.
- Includes login, booking, payment, notifications, and review features.

b. Integration Testing:

- Verifies the interactions between different modules, such as the connection between booking and payment systems or user notifications.
- Includes complex workflows like booking creation, payment processing, and notification delivery.

c. Boundary Value Analysis:

- Tests the application at the boundaries of input ranges to uncover edge case issues.

d. Equivalence Partitioning:

- Groups input data into valid and invalid equivalence classes for efficient testing.

e. Usability Testing:

- Validates that the user interface (UI) is intuitive and provides a smooth user experience.

B. Test Process

a. Requirement Analysis:

- Understand project requirements from functional specifications and user stories.
- Define acceptance criteria for each feature.

b. Test Planning:

- Develop a test plan outlining objectives, scope, resources, timelines, and deliverables.
- Identify tools and techniques for testing (e.g., Selenium, Postman).

c. Test Design:

- Create test cases for individual modules, integration workflows, and edge cases.
- Include boundary value and equivalence test scenarios.

d. Test Environment Setup:

- Configure a test environment that mirrors production conditions.
- Integrate the database, APIs, and cloud services for end-to-end testing.

e. Test Execution:

- Execute test cases manually and through automation tools.

f. Test Closure:

- Summarize testing activities, including the number of test cases executed, passed, failed, and blocked.
- Prepare a final test report with recommendations for improvement.

C. Workflow

a. Development Phase:

- Unit testing while developing.

b. QA Testing Phase:

- Execute planned functional and integration tests.

c. Bug Fixing and Verification:

- Defects are resolved and verified.
- Regression tests ensure no existing functionality is broken.

d. User Acceptance Testing (UAT):

- End-users test the application to validate real-world scenarios.

D. Methodologies Used

a. Agile Methodology:

- Testing is integrated into the development cycle, allowing continuous feedback and iterative improvements.
- Testing activities are planned for each sprint, ensuring timely validation of features.

b. Automation Testing:

- Automation tools like Selenium and JUnit are used for repetitive tasks such as regression and performance testing.

c. Black-Box Testing:

- Focuses on testing the application's functionality without knowledge of its internal code structure.

d. Risk-Based Testing:

- Prioritizes testing areas critical to user experience and business functionality, such as payments and booking modules.

e. Exploratory Testing:

- Testers explore the application to identify unexpected behavior and usability issues.

E. Key Tools and Technologies

a. Testing Tools: Selenium, Postman (for API testing).

b. Security Tools: Burp Suite.

1.1 Scope

A. In Scope

<u>Functional Requirements to be Tested:</u>

a. User Authentication and Management:

- User registration for vehicle owners and area providers.
- Login/logout functionality with secure password hashing.

b. Area Providing Module:

- Parking area registration with details (name, address, total spaces, price per hour, etc.).
- Editing, updating, and deleting parking area details.
- Real-time updates of available parking spaces.

c. Area Booking Module:

- Searching for parking areas based on geolocation and availability.
- Booking functionality with start and end times.
- Real-time updates to availability upon booking confirmation or cancellation.

d. Payment Processing:

- Integration with payment gateways for secure transactions.
- Payment methods: credit/debit cards, UPI, net banking.
- Payment status updates and refund mechanisms.

e. Notification System:

- Sending notifications for booking confirmations, cancellations, and payment updates.
- Marking notifications as read/unread.

f. Reviews and Ratings:

- Users can provide ratings and reviews for parking areas.
- Moderation and retrieval of reviews.

Non-Functional Requirements to be Tested:

a. Performance:

- Response time for booking searches and payment processing.
- Scalability to handle high user traffic during peak hours.

b. Usability:

- Intuitive user interface for both vehicle owners and area providers.

- Accessibility for diverse users, including responsive design for mobile devices.

c. Security:

- Data encryption for sensitive information (e.g., passwords, payment details).
- Protection against vulnerabilities like SQL injection, XSS, and CSRF.
- Secure API communication using HTTPS.

d. Compatibility:

- Cross-browser testing (e.g., Chrome, Firefox, Safari).
- Testing on multiple devices (Android, iOS, desktops).

e. Reliability:

- Ensuring system stability under various scenarios (e.g., high traffic, unexpected inputs).
- Database integrity with consistent updates to available spaces and user data.

f. Maintainability:

- Testing the modular structure for easy updates and fixes.
- Validation of automated test cases to facilitate future regression testing.

B. Out-of-Scope Testing

a. Hardware compatibility testing, as the platform relies on third-party devices for access.

1.2 Quality Objective

a. Conformance to Functional and Non-Functional Requirements:

Ensure the application adheres to the specified functional and non-functional requirements, including core features such as user authentication, area booking, payment processing, and notifications.

b. Delivering Quality as Defined by Stakeholders:

Validate that the Application Under Test (AUT) meets the quality standard

and expectations set by the client. This includes functionality, usability, performance, and security benchmarks to provide a seamless user experience.

c. Identifying and Resolving Bugs Before Deployment:

Proactively identify documents and resolve defects or issues in the application to ensure it is stable and error-free before going live.

d. User Satisfaction:

Guarantee that the platform delivers a satisfying experience to both vehicle owners and area providers, with intuitive workflows, responsive performance, and secure transactions.

e. System Reliability and Scalability:

Confirm that the platform can handle peak loads, large user bases, and extensive interactions between its modules without performance degradation.

f. Integration and Interoperability:

Validate the seamless integration of various modules (e.g., user management, booking, payments) and their ability to operate cohesively across different devices, browsers, and environments.

g. Data Integrity and Security:

Ensure that sensitive user data is securely stored, transmitted, and protected from unauthorized access or breaches.

h. Regulatory and Compliance Adherence:

To build trust with users and area providers, verify compliance with relevant legal, financial, and data protection standards (e.g., GDPR, PCI DSS).

i. Minimizing Risk:

Mitigate risks associated with functionality failures, security breaches, or performance issues by conducting comprehensive testing.

j. Readiness for Future Enhancements:

Lay a foundation for the maintainability and scalability of the application to accommodate future updates or feature additions seamlessly.

1.3 Roles and Responsibilities

A. QA Analyst (Quality Assurance Analyst)

- Role: Ensures the quality of the application through rigorous testing.
- Responsibilities:
    - Create and execute test cases based on functional and non-functional requirements.
    - Perform various types of testing such as functional, regression, usability, and performance testing.
    - Log and track defects in the defect management tool.
    - Validate defect fixes and perform retesting.
    - Collaborate with developers to understand issues and resolve them efficiently.
    - Prepare test summary reports and share findings with the team.

B. Developers

- Role: Build and maintain the application based on requirements and feedback.
- Responsibilities:
    - Develop features and functionalities as outlined in the project specifications.
    - Fix bugs identified during testing.
    - Perform unit testing and code reviews to ensure code quality.
    - Collaborate with QA analysts to understand and resolve defects.
    - Optimize the application for performance, security, and scalability.
    - Provide technical documentation for the developed modules.

C. UI/UX Designer

- Role: Designs the user interface and user experience to ensure a seamless user journey.
- Responsibilities:
    - Create wireframes, mockups, and prototypes based on requirements.
    - Ensure the design aligns with the target audience's preferences.
    - Collaborate with developers to ensure the design is implemented correctly.

- Conduct usability testing and make adjustments based on feedback.
- Maintain consistency in design elements throughout the application.

D.Database Administrator (DBA)

- Role: Manages the database to ensure data integrity, security, and performance.
- Responsibilities:
  - Design and implement the database schema.
  - Optimize database queries for performance.
  - Ensure data backup and recovery mechanisms are in place.
  - Monitor database health and resolve any issues.
  - Enforce security measures to protect sensitive data.

2 Test Methodology

2.1 Overview

Selected Methodology: Agile

Agile methodology was chosen for the Project project due to the following factors:

a. Nature of the Project

- Dynamic Requirements: The requirements for Project are expected to evolve based on user feedback, emerging trends, and stakeholder inputs. Agile allows for flexible adjustments during development.
- Module-Based Approach: The project is divided into modules (e.g., User Management, Area Booking, Payments, Notifications), which can be developed and tested incrementally.

b. Collaboration Needs

- Cross-Functional Teams: Agile promotes collaboration between developers, QA analysts, business analysts, and other stakeholders, which is essential for this multifaceted project

c. Focus on Quality

- Continuous Testing: Agile integrates testing throughout the development lifecycle, ensuring that bugs are identified and addressed early.
- User-Centric Development: By incorporating user feedback in every sprint, Agile ensures the application meets functional and non-functional requirements.

d. Risk Mitigation

- Frequent Releases: Regular releases reduce the risk of delivering a product that doesn't meet expectations.
- Early Issue Identification: With iterative development and testing, potential risks and issues are identified early, minimizing delays and cost overruns.

e. Project Complexity

- Integration of Multiple Modules: Agile supports the seamless integration of complex, interdependent modules like geolocation, payment systems, and notifications.
- High Scalability: Agile can accommodate the growing scope of features and functionalities as the project evolves.

f. Comparison with Other Methodologies

- Waterfall: Waterfall is rigid and doesn't accommodate changes well. For a dynamic project like Project, this would lead to delays and increased costs.
- Iterative: While iterative allows some flexibility, it lacks the collaboration and frequent feedback loops provided by Agile.
- Extreme Programming (XP): XP's focus on engineering practices might not align with the broader business and client collaboration needs of the project.

2.2 Test Levels

a. Unit Testing

- Purpose: To validate individual components or modules of the application in isolation.
- Scope:
    - Ensure that methods, functions, and classes for modules like User Management, Area Booking, and Payment Processing work as intended.
    - Test critical functions such as user authentication, vehicle registration, and booking creation.
- Responsible Team: Developers and QA
- Tools Used: JUnit (for Java-based testing).
- Example:
    - Verify that the validateUserCredentials () method correctly authenticates users.
    - Test if the calculateTotalCost () function calculates the booking cost accurately.

b. Integration Testing

- Purpose: To test interactions between modules and ensure seamless data flow.
- Scope:
    - Verify integrations between modules like:
        - User Management ↔ Area Booking.
        - Booking ↔ Payment.
        - Notifications ↔ Booking and Payments.
    - Ensure APIs and database queries are working correctly when modules communicate.
- Responsible Team: QA
- Tools Used: Postman (API Testing), Selenium (for end-to-end flows).

- Example:
    - Check if a successful booking triggers a notification.
    - Test payment success updates the booking status to "Confirmed."

c. Acceptance Testing

- Purpose: To validate the application meets client expectations and is ready for deployment.
- Scope:
    - Conduct User Acceptance Testing (UAT) with client and end-users.
    - Validate real-world scenarios, such as a user booking a parking spot at a specific location using their registered vehicle.
- Responsible Team: QA and End-User Representatives.
- Tools Used: Manual Testing, or tools such as BrowserStack for cross- platform compatibility testing.
- Example:
    - Validate if the area owner can see booking and payment details for their parking areas.

d. Regression Testing

- Purpose: To ensure new features or changes do not negatively affect existing functionality.
- Scope:
    - Rerun test cases for critical modules (e.g., User Management, Payment Processing) after introducing new features or fixes.
- Responsible Team: QA
- Tools Used: Selenium (for automation).
- Example:
    - Verify that adding a new payment method does not disrupt the booking workflow.

2.3 Test Completeness

a. Test Coverage

- Requirement:
  - Achieve 100% coverage for all critical modules and workflows.
  - All functional and non-functional requirements should be covered by test cases.
  - Verify edge cases, boundary conditions, and real-world scenarios.
- Verification:
  - Review test coverage reports to ensure every functionality and integration is tested

b. Execution of Test Cases

- Requirement:
  - Execute all manual and automated test cases planned during test design.
  - Ensure that tests for major workflows, such as user registration, booking, payment, and notification, are successfully executed.

- Verification:
  - Review the execution status in the test management tool (e.g., Jira, TestRail).
  - Confirm that test cases marked as critical and high priority are executed without errors.

c. Bug Resolution

- Requirement:
  - All critical and high-priority bugs identified during testing are resolved.
  - Medium and low-priority bugs should either be resolved or deferred with client approval for future releases.
- Verification:
  - No open critical bugs remain.

- Medium and low-priority issues are documented and scheduled for subsequent releases if not fixed.

d. Performance Benchmarks

- Requirement:
  - Meet all performance benchmarks, including response time, load capacity, and scalability.
  - Confirm that the application performs efficiently under peak load conditions.
- Verification:
  - Review performance testing reports from tools like JMeter to ensure all benchmarks are met.

e. Security Validation

- Requirement:
  - Address all identified vulnerabilities from penetration testing and security audits.
  - Ensure data encryption, secure payment processing, and proper access controls.
- Verification:
  - Review security test results and confirm no critical vulnerabilities remain unaddressed.

f. User Acceptance Testing (UAT)

- Requirement:
  - UAT feedback from the client and end-users is positive.
  - All reported issues during UAT are resolved or documented for future consideration.
- Verification:
  - Validate that UAT sign-off is obtained from stakeholders.

g. Regression Testing

- Requirement:
  - Complete regression testing to ensure that new changes have not affected existing functionalities.
- Verification:
  - Review the regression testing logs and confirm that no major issues were introduced during recent updates.

h. Documentation Completion

- Requirement:
  - All test artifacts, including test cases, defect reports, and test execution reports, are documented and shared with stakeholders.
  - User manuals and training documents are updated based on the final application.
- Verification:
  - Perform a final review of test documentation to ensure completeness and accuracy.

A. Test cases:

1. User Authentication Module

Test Case ID	Scenario	Input	Expected Output
UA-001	Login with valid	Valid	Success: User logged in
	credentials	username/password	
UA-002	Login with invalid credentials	Invalid username/password	Error: "Invalid credentials"

UA-003	Login with empty fields	Empty username/password	Error: "Fields cannot be empty"
UA-004	Password length boundary (min)	Password = 6 characters	Success: Logged in
UA-005	Password length boundary (max)	Password = 20 characters	Success: Logged in
UA-006	Account lock after 5 failed attempts	5 invalid login attempts	Error: "Account locked for security reasons"

2. Area Management Module

Test Case ID	Scenario	Input	Expected Output
AM-001	Add new area with valid details	Valid area details	Success: Area added
AM-002	Add new area with missing fields	Missing address/total spaces	Error: "All fields are required"
AM-003	Update area details (price per hour)	Valid price	Success: Details updated
AM-004	Update area with invalid price	Negative price	Error: "Invalid price value"
AM-005	Delete area with active bookings	Area with bookings	Error: "Cannot delete area with active bookings"

3. Booking Module

Test Case ID	Scenario	Input	Expected Output
BM-001	Book a parking space (valid)	Valid user, area, vehicle	Success: Booking created
BM-002	Book a space with invalid area ID	Invalid area ID	Error: "Area not found"
BM-003	Book space when no slots are available	Area with 0 available slots	Error: "No slots available"
BM-004	Cancel a booking (valid)	Booking ID	Success: Booking canceled
BM-005	Cancel a booking (expired)	Booking with past end time	Error: "Cannot cancel expired booking"

4. Payment Module

Test Case ID	Scenario	Input	Expected Output
PM-001	Process payment (valid)	Valid booking, amount	Success: Payment confirmed
PM-002	Process payment (failed gateway)	Valid details, gateway fails	Error: "Payment failed. Try again."
PM-003	Payment with partial amount	Amount < Total Cost	Error: "Insufficient payment amount"
PM-004	Payment refund on cancellation	Valid cancellation	Success: Amount refunded
PM-005	Payment using coupon	Valid coupon applied	Success: Discount applied,

	(valid)		payment processed

1. Notifications Module

Test Case ID	Scenario	Input	Expected Output
NT-001	Send booking confirmation notification	Valid booking	Notification sent to user
NT-002	Send notification on failed payment	Payment fails	Notification sent with retry instructions
NT-003	Notify area owner on low rating	Area rating drops below 3	Notification sent to owner
NT-004	Notification status update (read)	User reads notification	Status updated to "Read"

2. Reviews and Ratings Module

Test Case ID	Scenario	Input	Expected Output
RR-001	Add review with valid details	Valid area, rating, review text	Success: Review added
RR-002	Add review without rating	Missing rating	Error: "Rating is required"
RR-003	Add review with invalid rating	Rating > 5 or < 1	Error: "Rating out of range"
RR-004	Duplicate review submission	User submits a second review	Error: "Review already submitted"

3. Geolocation Module

Test Case ID	Scenario	Input	Expected Output
GL-001	Search parking near valid location	Latitude, Longitude	Success: List of parking areas returned
GL-002	Search parking with invalid location	Out-of-bounds coordinates	Error: "Location not supported"
GL-003	Calculate distance from user to area	Valid coordinates	Success: Distance calculated

B. Equivalence Testing:

1. User Authentication

Test Case ID	Input Field	Equivalence Classes	Expected Result
EP-001	Username	Valid: [1–50 chars], Invalid: [0, >50]	Valid: Accept; Invalid: Reject

EP-002	Password	Valid: [8–20 chars], Invalid: [<8, >20]	Valid: Accept; Invalid: Reject

2. Area Availability

Test Case ID	Input Field	Equivalence Classes	Expected Result
EP-003	Area Availability	Valid: Available slots > 0	Show available areas
EP-004		Invalid: Available slots = 0	Show "No slots available"

1. Payment

Test Case ID	Input Field	Equivalence Classes	Expected Result
EP-005	Card Number	Valid: 16 digits	Valid: Accept; Invalid: Reject
EP-006	Amount	Valid: [1–5000 INR], Invalid: [<1, >5000]	Valid: Accept; Invalid: Reject

C. Decision Table:

1. Decision Table for Complex Module Interactions

Condition	C1	C2	C3	C4	Outcome
Booking status is "Confirmed"	Yes	Yes	No	No	Notify user
Payment processed successfully	Yes	No	Yes	No	Update booking status to "Paid"
Area availability updated	Yes	Yes	Yes	No	Notify interested users

Expected Outcome	Notify	Deny	Update	Log Error	

2. Booking Cancellation and Refund

Condition	C1	C2	C3	C4	Action
Booking status is "Booked"	Yes	Yes	No	No	Process cancellation
Cancellation request within 1 hour	Yes	No	Yes	No	Refund full amount
Refund request outside 1 hour	No	Yes	No	Yes	Partial refund
Expected Outcome	Refund	No Refund	No Action	No Action	

3. Parking Availability Update

Condition	C1	C2	C3	C4	Action
Total spaces > 0	Yes	Yes	No	No	Allow booking
Available spaces > 0	Yes	No	Yes	No	Allow booking
Booking cancelled	No	Yes	No	Yes	Increase available spaces
Expected Outcome	Allow	Deny	Deny	Update	

3 Resource & Environment Needs

3.1 Testing Tools

a. Automation Tools

- Purpose: To automate repetitive test scenarios for regression, performance, and functional testing.
- Tools:
  - Selenium: For functional and regression testing of the web application.
  - Appium: For testing mobile versions of the application.
  - Postman: For API testing and validation.
  - JMeter: For performance and load testing.

b. Collaboration Tools

- Purpose: To facilitate communication and collaboration among team members.
- Tools:
    - Slack: For team communication.
    - Confluence: For maintaining centralized documentation and knowledge sharing.

c. Version Control Tool

- Purpose: To track changes to test scripts, automation code, and documentation.
- Tool:
    - Git: For version control and collaboration.
    - GitHub/: For hosting repositories.

d. Reporting Tools

- Purpose: To generate and share test execution and defect tracking reports.
- Tools:
    - Excel/Google Sheets: For simple reporting needs.

3.2 Test Environment

A. Hardware Requirements

a. Test Machines (Desktops/Laptops)

- Processor: Intel i5 or higher / AMD Ryzen 5 or higher
- RAM: Minimum 8 GB (16 GB recommended for automation and performance testing)
- Storage: Minimum 256 GB SSD (512 GB SSD or higher recommended for faster performance)
- Screen Resolution: 1920x1080 (Full HD) or higher
- Network: Stable broadband connection with minimum 50 Mbps speed

- Graphics: Integrated graphics (dedicated GPU if load testing requires graphical simulations)

b. Mobile Devices

- Android: Devices running Android 8 (Oreo) or higher
- Screen Sizes: A mix of devices covering small, medium, and large screen sizes for responsive testing

c. Servers (for Environment Hosting)

- CPU: 4-core or higher
- RAM: 16 GB or higher
- Disk Space: Minimum 1 TB (for database, logs, and backups)
- Network: High-speed Ethernet connection for local servers

B. Software Requirements

a. Operating System

- Windows 8 and above (recommended: Windows 10 or 11)
- macOS (for compatibility testing)
- Linux distributions (for backend testing and server environment)

b. Office Suite

- Microsoft Office 2013 and above
- Alternative: LibreOffice or Google Workspace

c. Email and Collaboration

- Microsoft Exchange for email management
- Slack, Microsoft Teams, or Zoom for team collaboration

d. Browsers (for Cross-Browser Testing)

- Chrome: Latest version
- Firefox: Latest version
- Microsoft Edge: Latest version
- Safari: Latest version (for macOS and iOS devices).

CHAPTER 7
CONCLUSION & FUTURE SCOPE

Conclusion:-

Urban mobility and traffic management are facing major issues due to the limited amount of land available and the growing demand for parking places in urban areas. This study has investigated a novel approach using a platform that maximizes unused parking spots. Bring open spaces and other private areas under one roof to create a scalable and sustainable alternative to parking infrastructure development. Our method exploits space efficiency by employing Internet of Things technology to provide real-time information on parking availability while reducing air pollution and traffic congestion caused by parking lot sailing. Because the platform's rating and review system ensures security in every transaction, trust is developed between owners of parking spaces and drivers of automobiles. Real estate owners have other direct financial opportunities. The goals of initiatives for smart cities and sustainable development encourage efficient use of existing resources rather than advocating for a large quantity of new buildings. By reducing traffic, the study suggests that managing vacant and unused spaces may be a long-term solution to parking problems and this will benefit both area owners and vehicle owners.

Future Scope:-

A. **Enhancing User Flow -** To enhance user flow, streamline the user interface by eliminating unnecessary steps and ensuring intuitive navigation. Clear call-to-action (CTA) buttons should be strategically placed for easy access, and visual cues like contrasting colors or animations can guide users through key actions.

B. **Dynamic Payment Flow -** A dynamic payment flow should offer users a variety of payment methods, from traditional credit cards to digital wallets and region-specific options like mobile payments. Localized payment solutions ensure the platform accommodates users' preferences based on their location.

C. **Enhancing Geo-Location Search -** Improving geo-location search involves offering advanced filters that allow users to refine their search based on proximity, such as adjusting the radius to find relevant results within a certain distance. Real-time location updates, utilizing GPS or IP-based tracking, automatically adjust search results based on the user's current position. Integrating interactive maps lets users visualize search results and navigate routes more effectively. Personalized, location-based recommendations, such as nearby events or services, enhance the experience, while geofencing allows for targeted notifications when users enter specific areas, offering deals or updates related to their location.

D. **Scalability and Expansion -** The platform's potential to reach entire cities, regions, and even nations is highly encouraging. As smart city technology becomes more widely used, it will be able to expand to new metropolitan regions and solve parking issues in urban areas across the globe.

E. **Integration with Public Transport Systems -** To make commuting easier and more connected, the platform could be expanded to integrate with public transport systems. This means users would be able to find parking spots near bus stops, train stations, or metro hubs. As a result, people might be more encouraged to park their vehicles and switch to public transport for the rest of their journey—reducing traffic and promoting a more sustainable way to travel in cities.

F. **Data-Driven Insights** - It has the potential to develop into a platform that offers city planners and local governments insightful information on urban mobility thanks to its access to vast amounts of real-time data. This information could improve the design of the next infrastructure projects, ease traffic, and reduce congestion.

G. **Partnerships with Commercial Establishments** - The number of parking spaces might be further increased by extending the platform's scope to include partnerships with commercial entities (such as shopping centres, hotels, and office buildings). This would provide consumers with more alternatives while assisting companies in making additional cash.

H. **Sustainability and Green Initiatives** - It can be integrated with programs which are sustainable to support eco-friendly parking solutions such as designating spaces for electric cars (EVs) with charging stations and promoting carpooling.

REFERENCES

[1] Parking Problems in INDIA: Measuring the Momentary Loss by Mr. Jovit Manjaly and Dr. Sebastian Tharapil Joseph. Parking Problems In India - Measuring The Monetary Loss.

[2] Issues and challenges of parking in Indian metropolises by Enrique Peñalosa. Issues and Challenges of Parking in Indian Metropolises

[3] Evaluation of Parking Characteristics: A case study of Delhi Janak Parmara, Pritikana Dasb, Farhat Azadb, Sanjay Davea, Ravindra Kumar. (PDF) Evaluation of Parking Characteristics: A case study of Delhi

[4] The issue of addressing the lack of parking spaces for road transport in cities by Jan Lizbetin and Ladislav Bartuska. The issue of addressing the lack of parking spaces for road freight transport in cities - a case study | Semantic Scholar

[5] Car Parking Problem in Urban areas, causes and Solutions by Hossam El-Din I. S. Ahmed, (PhD, Associate Professor Housing & Building National Research Center (HMRC)). Car Parking Problem in Urban Areas, Causes and Solutions by H. Ibrahim :: SSRN

[6] The Smart Parking Management System by Amira. A. Elsonbaty and Mahmoud Y. Shams. (PDF) The Smart Parking Management System

[7] Review Paper on Smart Parking System by Anusha, Arshitha, Anushree.Review-Paper-on-Smart-Parking-System-IJERTCONV7IS08017.pdf

[8] IoT-based Smart Parking System by Abhirup Khanna and Rishi Anand. untitled

[9] Smart Parking System by Suraj Kumar, Sunny Kumar Singh, Mr. Suman Jha. Smart Parking System

[10] Car Parking Problem and Solutions by Utkarsh Kumar, Vijendra Raj. The Smart Parking Management System | Elsonbaty | AIRCC's International Journal of Computer Science and Information Technology

[11] A Research on Smart Vehicle Parking System by Dipali Balmiki, Manas Singhal, Anupama Singh, Divyangi Tyagi. 1I43-IJSRMS0405740-v4-i7-pp124-127.pdf

[12] Parking Availability Prediction for Sensor-Enabled Car Parks in Smart Cities by Micans Infotech. (PDF) Parking Availability Prediction for Sensor-Enabled Car Parks in Smart Cities | Micans Infotech - Academia.edu

[13] Emerging Trends in Parking Report on a survey conducted by the 25_375x11_Outside_EmergingTrends_print

[14] **(2023)** Research Paper on Smart Car Parking System by Randive Sonali B.M. Polytechnic Belati, Solapur, Sakshi Kharade, Mhamane Srusti, Jadhav Sonali, Burhan Aliza. Research Paper on Smart Car Parking System – IJSREM

[15] Traffic management systems: A classification, review, challenges, and future perspectives by Allan M de Souza. Traffic management systems: A classification, review, challenges, and future

perspectives - Allan M de Souza, Celso ARL Brennand, Roberto S Yokoyama, Erick A Donato, Edmundo RM Madeira, Leandro A Villas, 2017

[16] **(2021)** Intelligent Traffic Management: A Review of challenges, solutions and future perspectives by Roopa Ravish1, Shanta Ranga Swamy. (PDF) Intelligent Traffic Management: A Review of Challenges, Solutions, and Future Perspectives

[17] Smart Traffic Management System by Ninad Lanke and Sheetal Koul. (PDF) Smart Traffic Management System

[18] **(2021)** Intelligent Traffic Management System Based on the Internet of Vehicles (IoV) by Samir A. Elsagheer Mohamed, Khaled A. AlShalfan. Intelligent Traffic Management System Based on the Internet of Vehicles (IoV) - Elsagheer Mohamed - 2021 - Journal of Advanced Transportation - Wiley Online Library

[19] **(2021)** A comprehensive review of intelligent traffic management using machine learning algorithms by Yash Modi, Ridham Teli, Akshat Mehta, Konark Shah & Manan Shah. A comprehensive review on intelligent traffic management using machine learning algorithms | Innovative Infrastructure Solutions

[20] **(2022)** Comparative Study of Shortest Distance Calculation Techniques in IoT-Based Wireless Sensor Networks by Ramandeep Gill & Tarun Kumar Dubey.

PATENT PROOF

(12) PATENT APPLICATION PUBLICATION

(19) INDIA

(22) Date of filing of Application :30/11/2024

(21) Application No.202411094176 A

(43) Publication Date : 17/01/2025

(54) Title of the invention : SYSTEM AND METHOD FOR RESOLVING PARKING ISSUES USING SMART VEHICLE PARKING

(51) International classification	:H04L0067120000, G06Q0010020000, H04L0009400000, G08G0001010000, G08G0001140000
(86) International Application No Filing Date	:NA :NA
(87) International Publication No	: NA
(61) Patent of Addition to Application Number Filing Date	:NA :NA
(62) Divisional to Application Number Filing Date	:NA :NA

(71)**Name of Applicant :**

1)KIET Group of Institutions

Address of Applicant :Delhi-NCR, Meerut Rd Ghaziabad Uttar Pradesh India 201206 Ghaziabad ---------- ----------

Name of Applicant : NA

Address of Applicant : NA

(72)**Name of Inventor :**

1)Ishita Srivastava

Address of Applicant :Computer Science Department, KIET Group of Institutions, Delhi-NCR, Meerut Rd Ghaziabad Uttar Pradesh India 201206 Ghaziabad ---------- ----------

2)Gurpreet Kaur

Address of Applicant :Computer Science Department, KIET Group of Institutions, Delhi-NCR, Meerut Rd Ghaziabad Uttar Pradesh India 201206 Ghaziabad ---------- ----------

3)Kritika

Address of Applicant :Computer Science Department, KIET Group of Institutions, Delhi-NCR, Meerut Rd Ghaziabad Uttar Pradesh India 201206 Ghaziabad ---------- ----------

4)Dr. Rishabh Jain

Address of Applicant :Computer Science Department, KIET Group of Institutions, Delhi-NCR, Meerut Rd Ghaziabad Uttar Pradesh India 201206 Ghaziabad ---------- ----------

(57) Abstract :

The present invention provides system and method for resolving parking issues using smart vehicle parking that ensures efficiency by integrating Internet of Things (IoT) devices into the parking lots. These modules instantly update a web application with the availability of a place based on automatic determination. By doing this, traffic flow is improved overall and the amount of time spent looking for parking is decreased. Area owners have complete control over their properties and can handle reservations and cancel them as needed. Reviews and ratings are meant to ensure security and trust, allowing drivers and parking suppliers to assess one another prior to confirming a reservation. Verification and other similar processes strengthen security and ensure safe transactions. Figure 1

No. of Pages : 19 No. of Claims : 7

RESEARCH PAPER PROOF

2025 International Conference on Electronics and Computing, Communication Networking Automation Technologies : Submission (218) has been created.

1 message

Microsoft CMT <noreply@msr-cmt.org> Mon, 21 Apr, 2025 at 1:04 pm
To: ishita.2125cs1199@kiet.edu

Hello,

The following submission has been created.

Track Name: ICEC2NT2025

Paper ID: 218

Paper Title: System and method for resolving the parking issues using smart vehicle parking.

Abstract:
There is a significant disparity between the number of cars and parking spots in metropolitan areas, particularly during rush hour, which results in traffic congestion, obstructions, and inconvenience. Our method addresses these problems by uniting vacant areas—owned by individuals or businesses—under a unified, well-designed platform. These spaces can be registered for public use, making the most of existing resources without the need for additional infrastructure. This will help to alleviate parking shortages and support the objectives of sustainable development and smart cities. To ensure efficiency, we integrate Internet of Things (IoT) devices into the parking lots. These modules instantly update a web application with the availability of a place based on automatic determination. By doing this, traffic flow is improved overall and the amount of time spent looking for parking is decreased. Area owners have complete control over their properties and can handle reservations and cancel them as needed. Reviews and ratings are meant to ensure security and trust, allowing drivers and parking suppliers to assess one another prior to confirming a reservation. Verification and other similar processes strengthen security and ensure safe transactions. Ultimately, our technology provides landowners with a sustainable and scalable solution to urban parking issues, as well as an opportunity to earn additional revenue.

Created on: Mon, 21 Apr 2025 07:33:56 GMT

Last Modified: Mon, 21 Apr 2025 07:33:56 GMT

Authors:
- ishita.2125cs1199@kiet.edu (Primary)
- gurpreet.2125cs1094@kiet.edu
- kritika.2125cs1033@kiet.edu
- rishabh.jain2986@gmail.com

Secondary Subject Areas: Not Entered

Submission Files:
IoT-based Safe Parking System and Minimizing Roadblocks.pdf (1 Mb, Mon, 21 Apr 2025 07:31:26 GMT)

Submission Questions Response: Not Entered

Thanks,
CMT team.

IoT-based Safe Parking System and Minimizing Roadblocks

Ishita Srivastava
Department of Computer Science
KIET Group of Institutions, Delhi NCR
Ghaziabad, Uttar Pradesh
ishita.2125cs1199@kiet.edu

Gurpreet Kaur
Department of Computer Science
KIET Group of Institutions, Delhi NCR
Ghaziabad, Uttar Pradesh
gurpreet.2125cs1049@kiet.edu

Kritika
Department of Computer Science
KIET Group of Institutions, Delhi NCR
Ghaziabad, Uttar Pradesh
kritika.2125cs1033@kiet.edu

Rishabh Jain
Department of Computer Science
KIET Group of Institutions, Delhi NCR
Ghaziabad, Uttar Pradesh
rishabh.jain2986@gmail.com

Abstract—There is a significant disparity between the number of cars and parking spots in metropolitan areas, particularly during rush hour, which results in traffic congestion, obstructions, and inconvenience. Our method addresses these problems by uniting vacant areas—owned by individuals or businesses—under a unified, well-designed platform. These spaces can be registered for public use, making the most of existing resources without the need for additional infrastructure. This will help to alleviate parking shortages and support the objectives of sustainable development and smart cities. To ensure efficiency, we integrate Internet of Things (IoT) devices into the parking lots. These modules instantly update a web application with the availability of a place based on automatic determination. By doing this, traffic flow is improved overall and the amount of time spent looking for parking is decreased. Area owners have complete control over their properties and can handle reservations and cancel them as needed. Reviews and ratings are meant to ensure security and trust, allowing drivers and parking suppliers to assess one another prior to confirming a reservation. Verification and other similar processes strengthen security and ensure safe transactions. Ultimately, our technology provides landowners with a sustainable and scalable solution to urban parking issues, as well as an opportunity to earn additional revenue.

Keywords—Parking Space Management, Smart City Solutions, IoT Parking Modules, Land Utilization, Real-Time Parking Availability, Traffic Congestion Solutions.

I. INTRODUCTION

Car ownership is increasing at a rapid rate, there is an imbalance in metropolitan areas between the number of cars and parking spots. This inequality leads to expanding urban problems of safe and lawful parking, as well as traffic jams and concerns about the safety of vehicle owners and pedestrians. In places with high population densities, such as marketplaces, commercial districts, and popular tourist destinations, parking problems are particularly problematic due to space constraints, which further complicate urban mobility and highlight the urgent need for creative solutions. A significant element contributing to this issue is the imbalance between parking availability and demand. On-street parking, which is frequently encroaching on private property and reducing the amount of space on the road available for cars, is the primary cause of traffic. In addition, cruising—the act of driving about in search of a parking space- has become popular among drivers. This leads to traffic congestion, air pollution, and inefficient fuel consumption. The problem is worse by the absence of a conventional system for parking slots, which results in an inefficient use of available space because many parking spots are either too tiny or too huge. Additionally, driving stress and general road safety are decreased by drivers' ongoing fear of fines (challans) for parking illegally.

It is difficult to solve these parking issues without adding more infrastructure, particularly in big cities where there is a lack of available space. As an alternative to depending on pricey development projects to provide more parking spots, this attempts to offer a sustainable, scalable solution by making the most of already-existing resources. Our concept tackles parking overcrowding by integrating empty spaces, including idle commercial spaces and private properties, into a single parking platform. This approach not only addresses the present parking shortage but also advances the more general goals of smart city and sustainable development initiatives. It works as a middleman between car owners and parking space providers to deliver a safe, effective, and adaptable parking solution. Car owners may quickly find and reserve parking spots based on price, location, and availability by using an online platform.

Area	Fact	Source
Traffic Congestion	Urban areas face a **30% increase in congestion** due to vehicles searching for parking spaces.	International Parking Institute
Search Time	Drivers in India spend an average of **20 minutes searching for parking**.	Boston Consulting Group

Unused Spaces	Studies estimate that **20-30% of urban spaces remain unused**, even during peak hours.	Urban Land Institute
Economic Impact	Efficient parking solutions can save up to **₹1,500 crores annually** in wasted fuel and time costs.	McKinsey & Company
Industry Growth	The parking industry is expected to grow at a **CAGR of 9.6% from 2023 to 2028**.	Market Research Future
Vehicle Theft	**58% of urban vehicle thefts** occur in unregulated or unsafe parking spaces.	National Crime Records Bureau, India
IoT Effectiveness	IoT-based surveillance can reduce theft incidents by **40%**.	Frost & Sullivan
Space Utilization	Smart parking solutions powered by IoT can increase parking space utilization by **20-30%**.	Allied Market Research
Global Market Size	Global IoT in parking market size is projected to reach **$12 billion by 2025**.	Grand View Research
Environmental Impact	Idling vehicles searching for parking contributes to **10% of urban CO_2 emissions**.	Environmental Protection Agency, USA
Emission Reduction	Smart parking systems can reduce CO_2 emissions by up to **40%**.	Smart Cities Council
Internet Access	**95% of urban dwellers in India** have access to mobile internet, enabling digital parking solutions.	Telecom Regulatory Authority of India
Payment Gateway	Payment gateways like Razorpay handle **$50 billion in annual transactions**.	Razorpay Report, 2023
Smart Cities Mission	The **Smart Cities Mission** has allocated **₹6,000 crores** for smart urban infrastructure.	Ministry of Housing and Urban Affairs, India

Table 1: Impact on various areas listed

However, property owners can increase their income without making large investments by renting out the places they don't utilize. Parking spots will have Internet of Things (IoT) modules placed to give real-time availability updates, making it easy and quick for users to find and reserve spaces. In addition to lowering the need for cruising, the system's dynamic booking and availability tracking help enhance traffic flow and cut down on emissions from prolonged idling.

II. Literature Review

Due to rising vehicle ownership and urbanization, nearly 40 million vehicles are clogging India's urban centres, creating serious parking issues. Economic progress and population increase have made this problem worse. There are two types of parking issues: off-street parking, which is usually found in big businesses and retail centres, and on-street parking, which is more erratic and driven by the market and frequently causes traffic jams. The inability to incorporate parking regulations into urban planning has led to several problems, such as ineffective land use, poor transportation quality, and negative environmental effects. It is crucial to increase parking spaces, improve accessibility, and maximize current infrastructure to address these issues. The main answers are to promote public transit, use dynamic parking pricing, and apply smart pricing in commercial zones. The assessment of parking problems in India looks at the various aspects of the issue, which is fuelled by growing vehicle ownership and fast urbanization. This thorough research, which is based on several studies, emphasizes the necessity of addressing the parking space deficit and its effects on urban mobility. The number of vehicles and India's expanding population have put tremendous strain on the country's infrastructure. To maximize parking usage, creative solutions have been put forth, such as smart parking systems. Manjaly and Joseph [1] point out that problems with parking cost India's GDP 1.56% annually. Their suggested smart parking solutions provide effective space distribution, easing traffic and enhancing comfort. In a similar vein, parking management has found use in mobile applications. Parmar et al. [3] provide examples of how users can find and book parking spaces.

Paper reference no.	Year	Key finding	Research focus
25	2023	SCOPE model, utilizing AlexNet and YOLO-V3 as learning controllers, achieved high accuracy (99.87% and 99.89%, respectively) in identifying parking lot statuses, demonstrating its effectiveness for smart city ecosystems	Revolutionizing Urban Mobility: IoT-Enhanced Autonomous Parking Solutions with Transfer Learning for Smart Cities

24	2023	Arduino-based smart parking system offers a cost-effective and adaptable solution for managing parking demand, improving space availability and efficiency, with potential for scaling and enhancing smart city applications	IoT-Enabled Smart Parking: Enhancing Efficiency and Sustainability in Smart Cities
14	2023	Arduino-based smart parking system effectively enhances parking efficiency, reduces congestion and emissions, and supports sustainable urban development through affordable, scalable IoT integration	Research Paper on Smart Car Parking System
20	2022	IoT-enabled wireless sensor networks, simulated using CupCarbon IoT 5.0, effectively calculate the shortest distance between nodes through comparative analysis of routing techniques	Comparative Study of Shortest Distance Calculation Techniques in IoT-Based Wireless Sensor Networks
16	2021	Intelligent Transportation Systems (ITS), leveraging wireless networks, deep learning, and optimization algorithms, effectively reduce traffic congestion and improve travel time when tailored to location-specific needs	Intelligent Traffic Management : A Review of challenges, solutions and future perspectives
19	2021	AI and ML-driven traffic management systems optimize traffic flow, reduce congestion, and minimize manual intervention through advanced algorithms and smart technologies	A comprehensive review of intelligent traffic management using machine learning algorithms
23	2021	ILP and FIS-based parking management model can optimize campus parking assignments, reduce wandering time, and enhance satisfaction by considering user preferences and institutional priorities	Parking Demand vs Supply: An Optimization-Based Approach at a University Campus
1	2019	well-managed and accessible parking system is essential to alleviate parking shortages, reduce traffic congestion, enhance convenience, and provide social, environmental, and governmental benefits	Parking Problems in INDIA: Measuring the Momentary Loss
5	2017	modern technological solutions and improved urban planning are essential to address the imbalance between parking supply and demand, while also considering environmental impacts for more livable cities.	Car Parking Problem in Urban areas, causes and Solutions

Table 2: Literature review summary

Furthermore, Peñalosa [2] emphasizes how important it is to match parking regulations with urban planning. Parking

supply, price, and enforcement laws are largely established by municipal authorities. Policies such as zoning laws, parking requirement requirements for new construction, and dynamic pricing structures are useful tools for encouraging economical land use and lowering the ownership of excessive numbers of vehicles. As Lizbetin and Bartuska [4] have observed, the scarcity of parking spots for road freight transit is another major problem. Urban mobility plans that incorporate walking, bicycling, and public transportation can lessen reliance on private vehicles and thereby ease the demand for parking. It's also important to learn more about how the public views parking and behaves about it.

To effectively create solutions to solve parking issues, Ahmed [5] emphasizes the significance of comprehending travel patterns and preferences. Sustainable environmental practices are also a major component of parking management. Numerous studies highlight how crucial electric vehicle charging stations and green parking infrastructure are to minimizing parking's negative environmental effects. Furthermore, as Amira Elsonbaty and Mahmoud Shams have discussed, cloud computing and IoT have improved smart parking solutions [6]. Real-time tracking of available parking spaces is made possible by their Smart Parking Management System (SPMS), which makes use of Arduino, Internet of Things sensors, and mobile applications.

The parking availability is tracked by the system using infrared sensors. Data is sent to a server over Wi-Fi and accessed by users through a mobile app. This app offers an affordable, easy-to-use platform for finding open places, booking them, and getting updates. By utilizing IoT, SPMS streamlines parking administration and lowers fuel usage by cutting down on the amount of time spent looking for parking spots. Furthermore, Khanna and Anand's work [7] describes an Internet of Things (IoT)-based cloud-integrated parking system that uses sensors to keep an eye on every parking spot and notifies users via a mobile application of its real-time availability. This kind of solution eases traffic congestion and improves urban mobility by drastically cutting down on the amount of time wasted looking for parking spots.

PPPs, or public-private partnerships, are also essential for funding and constructing parking infrastructure. To mobilize resources and scale up parking solutions, Ahmed [5] promotes cooperative arrangements between public authorities, commercial developers, and technological companies. Building consensus on parking management solutions requires including stakeholders and the community in decision-making processes.

In summary, parking issues in India's densely populated cities are complex and need an all-encompassing solution. Novel approaches to parking management are required in light of problems like oversaturated parking spots, unregulated rates, and environmental issues. Effective legislation, public awareness initiatives, and technology developments should support these efforts. India can reduce traffic congestion, improve the quality of urban life, and lessen its environmental effects by taking a comprehensive approach to parking difficulties.

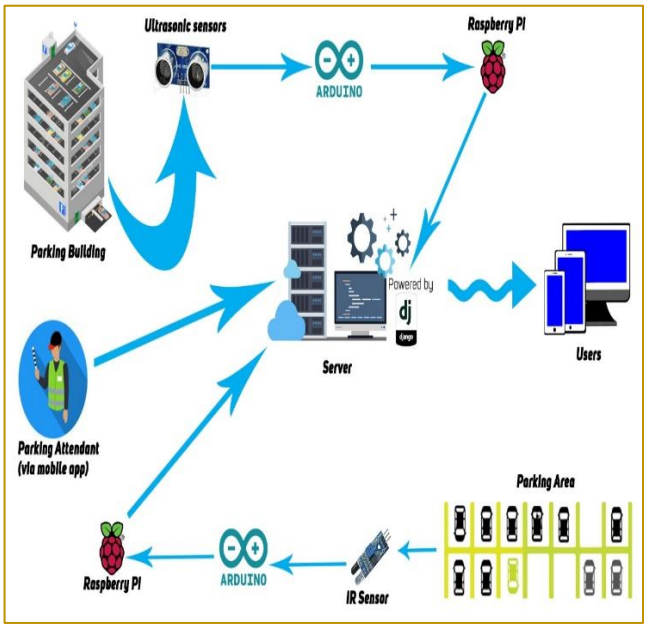


Fig.1. IoT-based smart parking system [21]

A. Existential Problem

- **Underutilization of Existing Parking Spaces:** To address the rising demand for parking, the majority of research and solutions centre on developing additional infrastructure. Optimizing the utilization of underutilized areas, such as garages, private plots, and other vacant assets, is noticeably lacking in attention. There aren't many alternatives that combine these areas into a centralized system that facilitates simple public parking access.
- **Lack of Real-Time Availability Data:** Users must drive about looking for parking spaces because many of the parking solutions on the market today do not offer real-time data on the availability of spots. This causes pollutants in addition to causing traffic congestion. Although smart parking has been the subject of various studies, little research has been done on how to integrate this technology in poor nations in a way that is both scalable and affordable.
- **Inadequate Focus on Sustainable Development:** Although there is a growing global concern about sustainable urban development, many parking systems just consider convenience and ignore the environmental impact of adding more infrastructure. The body of research on strategies that support sustainable development objectives by making the most of already-existing spaces and minimizing the need for new construction is lacking.
- **Limited Community Engagement Models:** The potential of community engagement is largely overlooked in current parking management research. There aren't many studies that address how private citizens or companies can add empty spots to a broader parking ecosystem, resulting in a win-win situation. There aren't many outlets that promote community involvement in resolving urban parking issues.

- **Lack of Flexible and Transparent Booking Systems:** Numerous parking systems now in use do not provide flexibility or customized booking alternatives depending on user preferences like cost, distance, or location. Furthermore, few studies have addressed the need for user-friendly platforms that provide safe payment methods and clear information on parking rates. Transparency in booking and pricing is still a big challenge.
- **Insufficient Safety and Trust Mechanisms:** Rarely does the literature now in publication address the safety worries of both car owners and space providers. Few studies have offered comprehensive procedures like a rating and review system, paired with verification processes to assure confidence and security in parking transactions. This is a serious imbalance, particularly in crowded places where people prioritize their safety.
- **Challenges with Scalability and Implementation in Developing Countries:** Although several smart parking solutions have been put into practice in affluent nations, there is still a lack of research on how scalable these solutions would be in highly crowded, resource-constrained cities. Studies that have been done in the past frequently undervalue the particular economic, social, and infrastructure problems that emerging nations face, which makes it difficult to apply or scale up beneficial solutions in these areas.

Region	Vacancy Rate	Vacant Units (millions)	Key Cities/Areas	Key Observations
Urban India (Overall)	11.1%	11.1 million	Across cities	Due to high investment properties and poor land-use planning.
Greater Noida (U.P.)	61%	Approx. 100,000 units	Greater Noida, surrounding NCR areas	Significant portion of housing stock remains unoccupied.
Mumbai (Maharashtra)	15%	Approx. 300,000 units	Mumbai Metropolitan Region	15% vacant housing, mostly in peripheral areas.
Vasai-Virar (Maharashtra)	50%	Approx. 70,000 units	Vasai-Virar	Large vacant properties due to overbuilt areas for investment.
Jammu & Kashmir	20%	Approx. 100,000 units	Srinagar, Jammu	High vacancy due to migration and demographic shifts.
Gujarat	18%	Approx. 250,000 units	Ahmedabad, Surat, Vadodara	Excess housing in rapidly developing industrial areas.

Table 3: Vacant and unused spaces in India [27] [28]

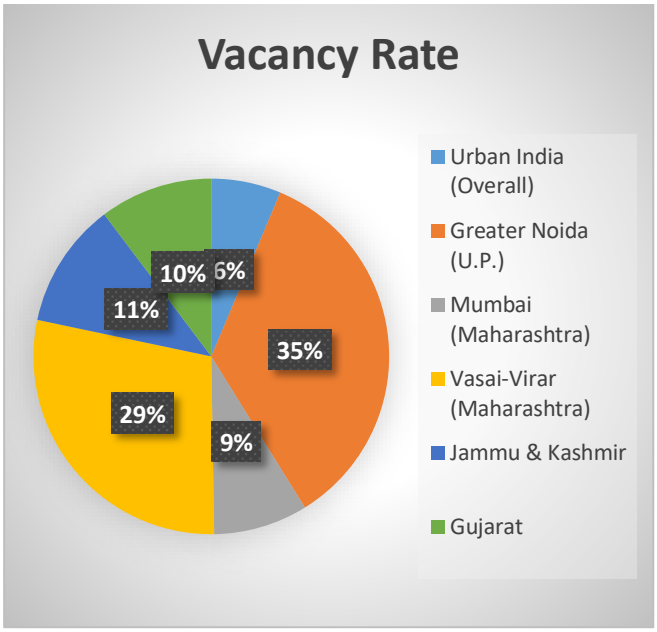


Fig.2. Vacant and unused spaces in India

III. PROPOSED SYSTEM

The goal of the proposed system is to simplify parking administration by combining open spots onto a single platform. This system uses a centralized web platform, IoT integration, and real-time data to enable safe, adaptable, and effective parking options for both car owners and area providers. The following is an overview of the system's main elements, including its functional modules and database structure.

A. Centralized Platform and User Management

- The web-based platform that acts as a middleman between area suppliers and car owners is the central component of the system. Users can register, search, and manage parking spaces, whether they are vehicle or area owners.
- Users Table: This table stores essential information about users, including vehicle owners and area owners.

It records user credentials and contact details, ensuring user verification and profile management.

- Vehicles Table: For vehicle owners, this table stores information about their registered vehicles, linking them to their respective user accounts for efficient vehicle management.

B. Parking Space Registration and Management

- Parking space owners can register their spaces on the platform, and car owners can then book those spaces. Through IoT modules placed in the parking lots, the system enables real-time updates on parking availability.
- Areas Table: This table stores parking area details, including location (latitude and longitude), pricing, total and available spaces. It links parking areas to area owners and ensures real-time availability updates.
- IoT Integration: IoT modules will be installed in parking areas to detect the presence of vehicles, allowing for real-time monitoring of parking space availability. This data will be automatically updated on the web platform.

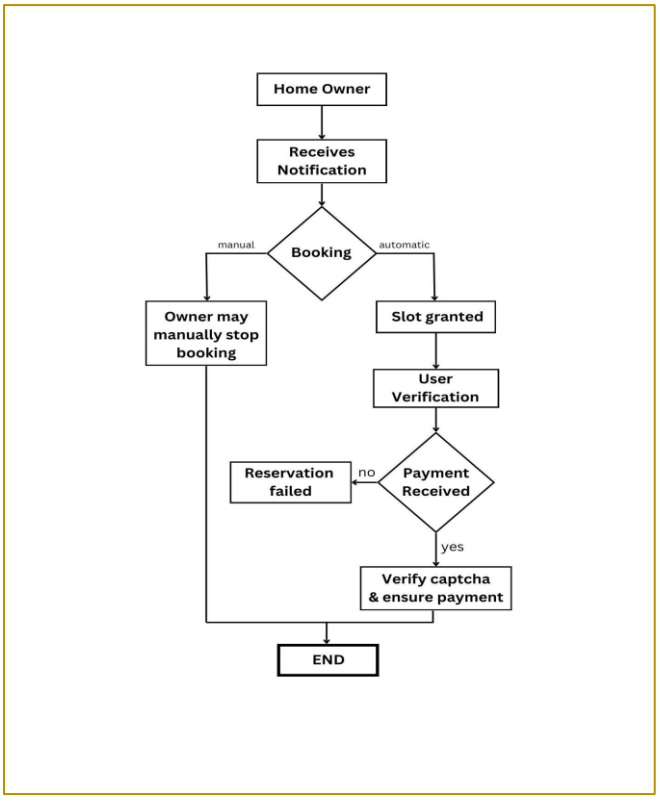


Fig.3. Area owner's data flow diagram

C. Secure Payment Processing

- A payment gateway is integrated into the system to handle transactions. Both area owners and users can monitor their earnings and securely make payments for their reservations.
- Payments Table: This table records payment transactions linked to bookings, ensuring secure and transparent financial operations. Payment methods and transaction statuses (e.g., completed or pending) are also recorded.
- Payment Integration: The system follows industry-standard payment protocols to safeguard transactions and ensure data security for both vehicle owners and area providers.

D. Notification and Alerts System

- The platform's integrated notification system makes sure users are informed of any updates, payment status, and booking confirmations.
- Notifications Table: This table stores notifications sent to users, ensuring they receive timely updates about bookings, payments, and platform changes. Notifications are linked to user profiles, ensuring personalised communication.

E. Real-Time Parking Booking and Management

- Using the platform, vehicle owners may look for, pick, and reserve parking spots. Users can customize their options using the booking module according to criteria like availability, cost, and proximity.

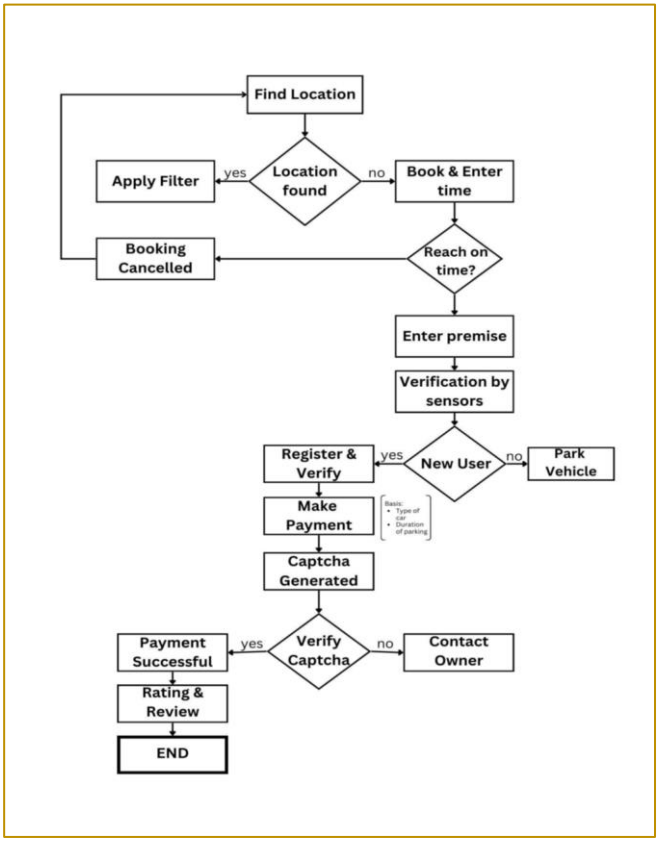


Fig.4. Vehicle owner's data flow diagram

- Bookings Table: This table records parking bookings, linking vehicle owners, their vehicles, and the selected parking area. It tracks booking details, including start and end times, costs, and status (booked, completed, or cancelled).
- Dynamic Space Management: Upon a confirmed booking, the system automatically updates the

available spaces in the Areas Table, ensuring that real-time data is reflected on the platform.

F. Review and Rating System

- The platform lets customers rank and write reviews about their parking experiences, which improves trust and safety. The quality of services for both car owners and local suppliers is enhanced by these comments.

IV. IMPLEMENTATION

We used an Arduino, an ESP8266, and a MQTT-based system to develop an Internet of Things solution that uses location coordinates to calculate the distance between two sites. Latitude and longitude (GPS coordinates) had to be obtained from two different sites for the setup to work. The distance between these coordinates was computed in the code using the Haversine formula [22]:

$$d = 2r \arcsin\left(\sqrt{\sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + \cos(\phi_1)\cos(\phi_2)\sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \quad (1)$$

ϕ1 and ϕ2 stand for the latitudes; Δϕ is the difference between the latitudes and Δλ is the difference between the longitudes. The symbol r signifies the Earth's radius or around 6371 km. We set up the Arduino to record GPS data, compute distance using the Haversine method, and display the result. The system was outfitted with an ESP8266 for wireless connectivity, enabling the instantaneous transmission of the calculated distance to an MQTT broker. This made it possible to track and monitor distances between two places remotely using the Internet of Things platform.

Algorithm:

```
// Java program for the haversine formula
public class HaversineFormula {
        static double haversine (double lat1,
double lon1, double lat2, double lon2) {
// distance between latitudes and longitudes
        double disLat = Math.toRadians(lat2 - lat1);
        double disLon = Math.toRadians(lon2 - lon1);
// convert to radians
        lat1 = Math.toRadians(lat1);
        lat2 = Math.toRadians(lat2);
// apply formulae
        double n = Math.pow(Math.sin(disLat / 2), 2) +
Math.pow(Math.sin(disLon / 2), 2) * Math.cos(lat1) *
Math.cos(lat2);
        double rad = 6371;
        double m = 2 * Math.asin(Math.sqrt(n));
        return rad * m;
    }
    // Driver Code
    public static void main(String[] args){
        double lat1 = 28.741023;
        double lon1 = 77.500771;
        double lat2 = 28.828984;
        double lon2 = 77.577019;
        System.out.println (haversine(lat1, lon1, lat2, lon2)
    + " K.M.");
    }
}
```

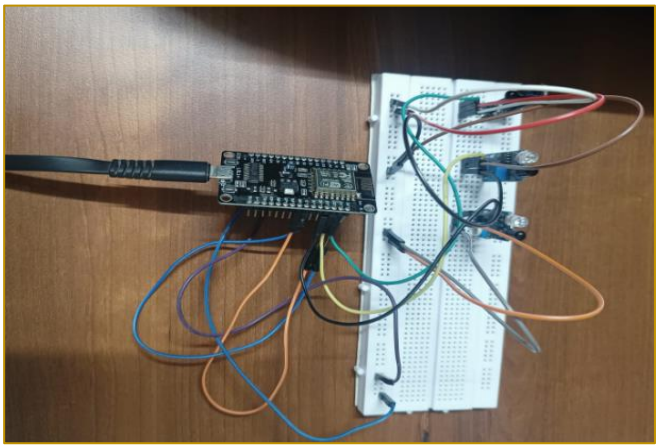


Fig.5. IoT implementation

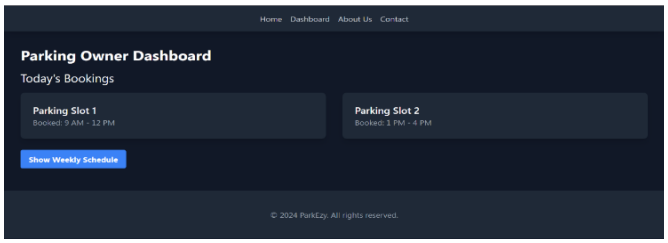


Fig.6. Available Parking slots showing on our website

Proof of the formula [26]:

$$\text{hav}(\theta) = \text{hav}(\Delta\varphi) + \cos(\varphi_1)\cos(\varphi_2)\,\text{hav}(\Delta\lambda) \quad (2)$$

by transforming the points given by their latitude and longitude into cartesian coordinates, then taking their dot product. Consider two points p1, p2 on the unit sphere, given by their latitude φ and longitude λ :

$$\mathbf{p_2} = (\lambda_2, \varphi_2)$$
$$\mathbf{p_1} = (\lambda_1, \varphi_1) \quad (3)$$

These representations are very similar to spherical coordinates; however, latitude is measured as angle from the equator and not the north pole. These points have the following representations in cartesian coordinates:

$$\mathbf{p_2} = (\cos(\lambda_2)\cos(\varphi_2),\ \sin(\lambda_2)\cos(\varphi_2),\ \sin(\varphi_2))$$
$$\mathbf{p_1} = (\cos(\lambda_1)\cos(\varphi_1),\ \sin(\lambda_1)\cos(\varphi_1),\ \sin(\varphi_1)) \quad (4)$$

From here we could directly attempt to calculate the dot product and proceed, however the formulas become significantly simpler when we consider the following fact: the distance between the two points will not change if we rotate the sphere along the z-axis. This will in effect add a constant to λ1, λ2. Note that similar considerations do not apply to transforming the latitudes - adding a constant to the latitudes may change the distance between the points. By choosing our constant to be -λ1, and setting λ`=λ, our new points become:

$$\mathbf{p}'_2 = (\cos(\lambda')\cos(\varphi_2),\ \sin(\lambda')\cos(\varphi_2),\ \sin(\varphi_2))$$
$$\mathbf{p}'_1 = (\cos(0)\cos(\varphi_1),\ \sin(0)\cos(\varphi_1),\ \sin(\varphi_1))$$
$$= (\cos(\varphi_1),\ 0,\ \sin(\varphi_1)) \quad (5)$$

With θ denoting the angle between p1 and p2, we now have that:

$$\cos(\theta) = \langle \mathbf{p}_1, \mathbf{p}_2 \rangle = \langle \mathbf{p}'_1, \mathbf{p}'_2 \rangle = \cos(\lambda')\cos(\varphi_1)\cos(\varphi_2) + \sin(\varphi_1)\sin(\varphi_2)$$
$$= \sin(\varphi_2)\sin(\varphi_1) + \cos(\varphi_2)\cos(\varphi_1) - \cos(\varphi_2)\cos(\varphi_1) + \cos(\lambda')\cos(\varphi_2)\cos(\varphi_1)$$
$$= \cos(\Delta\varphi) + \cos(\varphi_2)\cos(\varphi_1)(-1 + \cos(\lambda')) \Rightarrow$$
$$\operatorname{hav}(\theta) = \operatorname{hav}(\Delta\varphi) + \cos(\varphi_2)\cos(\varphi_1)\operatorname{hav}(\lambda') \quad (6)$$

This proof has been cited from Wikipedia [26] and used in this project.

Below are the tables with calculated distances:

Area	Latitude	Longitude
SRM NCR	28.741023	77.500771
Modinagar Temple	28.828984	77.577019
Meerut City Station	28.976200	77.703700
Ghaziabad Junction	28.669156	77.453758
Hindon Airport	28.665617	77.432001
India Gate, Delhi	28.612894	77.229446
Taj Mahal, Agra	27.175144	78.042142
Jaipur Railway Station	26.919650	75.787819

Table 4: Areas with Corresponding Latitude and Longitude

From	To	Distance (km)
SRM NCR	Modinagar Temple	12.6
SRM NCR	Meerut City Station	35.0
Ghaziabad Junction	Hindon Airport	3.0
India Gate, Delhi	Taj Mahal, Agra	194.2
Taj Mahal, Agra	Jaipur Railway Station	233.7

Table 5: Calculated distances for parking using the Haversine formula

A. Technology used

- **Hardware:** consists of IoT sensors, placed in parking slots to identify availability status and provide data in real-time using Wi-Fi networks to a central server. Then it transmits data to a cloud like AWS. Gateway devices are also needed for connecting different modules. Additionally, networking hardware such as modems and routers are needed for connectivity. To engage with platform and parking slots, users must have smartphones or computers with proper internet connectivity.

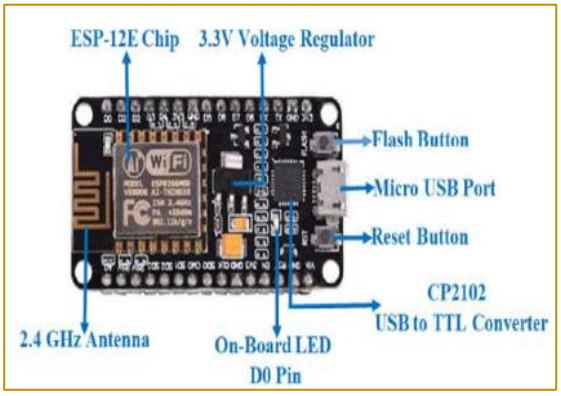


Fig.7. Node MCU ESP8266 development board [6]

Fig.8. Arduino UNO board [6]

- **Software:** comprises of web-based platform with Django for backend and React framework for frontend. IoT sensor is needed for real-time data processing, which is controllable using MQTT protocols. AWS is needed for hosting, storage and scaling of the system. Bookings, parking slot availability and user data are managed using PostgreSQL and MySQL. SSL encryption provides secure transactions and data integrity.

B. Future Scope:

- **Scalability and Expansion**: The platform's potential to reach entire cities, regions, and even nations is highly encouraging. As smart city technology becomes more widely used, it will be able to expand

to new metropolitan regions and solve parking issues in urban areas across the globe.

- **Integration with Public Transport Systems**: To allow for integration with public transport systems, the platform might be expanded. This would allow users to find parking near bus stops, train stations, and metro hubs. This will lead to a decline in the use of private vehicles for urban commuting and an increase in the use of public transportation.
- **Data-Driven Insights**: It has the potential to develop into a platform that offers city planners and local governments insightful information on urban mobility thanks to its access to vast amounts of real-time data. This information could improve the design of the next infrastructure projects, ease traffic, and reduce congestion.
- **Partnerships with Commercial Establishments**: The number of parking spaces might be further increased by extending the platform's scope to include partnerships with commercial entities (such as shopping centres, hotels, and office buildings). This would provide consumers with more alternatives while assisting companies in making additional cash.
- **Sustainability and Green Initiatives**: It can be integrated with programs which are sustainable to support eco-friendly parking solutions such as designating spaces for electric cars (EVs) with charging stations and promoting carpooling. This also aligns with global goals aimed at reducing traffic while providing more sustainable urban ecosystems.

V. Conclusion

Urban mobility and traffic management are facing major issues due to the limited amount of land available and the growing demand for parking places in urban areas. This study has investigated a novel approach using a platform that maximizes unused parking spots. Bring open spaces and other private areas under one roof to create a scalable and sustainable alternative to parking infrastructure development. Our method exploits space efficiency by employing Internet of Things technology to provide real-time information on parking availability while reducing air pollution and traffic congestion caused by parking lot sailing. Because the platform's rating and review system ensures security in every transaction, trust is developed between owners of parking spaces and drivers of automobiles. Real estate owners have other direct financial opportunities. The goals of initiatives for smart cities and sustainable development encourage efficient use of existing resources rather than advocating for a large quantity of new buildings. By reducing traffic, the study suggests that managing vacant and unused spaces may be a long-term solution to parking problems and this will benefit both area owners and vehicle owners.

VI. References

[1] Parking Problems in INDIA: Measuring the Momentary Loss by Mr. Jovit Manjaly and Dr. Sebastian Tharapil Joseph. Parking Problems In India - Measuring The Monetary Loss.

[2] Issues and challenges of parking in Indian metropolises by Enrique Peñalosa. Issues and Challenges of Parking in Indian Metropolises

[3] Evaluation of Parking Characteristics: A case study of Delhi Janak Parmara, Pritikana Dasb, Farhat Azadb, Sanjay Davea, Ravindra Kumar. (PDF) Evaluation of Parking Characteristics: A case study of Delhi

[4] The issue of addressing the lack of parking spaces for road transport in cities by Jan Lizbetin and Ladislav Bartuska. The issue of addressing the lack of parking spaces for road freight transport in cities - a case study | Semantic Scholar

[5] Car Parking Problem in Urban areas, causes and Solutions by Hossam El-Din I. S. Ahmed, (PhD, Associate Professor Housing & Building National Research Center (HMRC)). Car Parking Problem in Urban Areas, Causes and Solutions by H. Ibrahim :: SSRN

[6] The Smart Parking Management System by Amira. A. Elsonbaty and Mahmoud Y. Shams. (PDF) The Smart Parking Management System

[7] Review Paper on Smart Parking System by Anusha, Arshitha, Anushree.Review-Paper-on-Smart-Parking-System-IJERTCONV7IS08017.pdf

[8] IoT-based Smart Parking System by Abhirup Khanna and Rishi Anand. untitled

[9] Smart Parking System by Suraj Kumar, Sunny Kumar Singh, Mr. Suman Jha. Smart Parking System

[10] Car Parking Problem and Solutions by Utkarsh Kumar, Vijendra Raj. The Smart Parking Management System | Elsonbaty | AIRCC's International Journal of Computer Science and Information Technology

[11] A Research on Smart Vehicle Parking System by Dipali Balmiki, Manas Singhal, Anupama Singh, Divyangi Tyagi. 1I43-IJSRMS0405740-v4-i7-pp124-127.pdf

[12] Parking Availability Prediction for Sensor-Enabled Car Parks in Smart Cities by Micans Infotech. (PDF) Parking Availability Prediction for Sensor-Enabled Car Parks in Smart Cities | Micans Infotech - Academia.edu

[13] Emerging Trends in Parking Report on a survey conducted by the 25_375x11_Outside_EmergingTrends_print

[14] **(2023)** Research Paper on Smart Car Parking System by Randive Sonali B.M. Polytechnic Belati, Solapur, Sakshi Kharade, Mhamane Srusti, Jadhav Sonali, Burhan Aliza. Research Paper on Smart Car Parking System – IJSREM

[15] Traffic management systems: A classification, review, challenges, and future perspectives by Allan M de Souza. Traffic management systems: A classification, review, challenges, and future perspectives - Allan M de Souza, Celso ARL Brennand, Roberto S Yokoyama, Erick A Donato, Edmundo RM Madeira, Leandro A Villas, 2017

[16] **(2021)** Intelligent Traffic Management: A Review of challenges, solutions and future perspectives by Roopa Ravish1, Shanta Ranga Swamy. (PDF) Intelligent Traffic Management: A Review of Challenges, Solutions, and Future Perspectives

[17] Smart Traffic Management System by Ninad Lanke and Sheetal Koul. (PDF) Smart Traffic Management System

[18] **(2021)** Intelligent Traffic Management System Based on the Internet of Vehicles (IoV) by Samir A. Elsagheer Mohamed, Khaled A. AlShalfan. Intelligent Traffic Management System Based on the Internet of Vehicles (IoV) - Elsagheer Mohamed - 2021 - Journal of Advanced Transportation - Wiley Online Library

[19] **(2021)** A comprehensive review of intelligent traffic management using machine learning algorithms by Yash Modi, Ridham Teli, Akshat Mehta, Konark Shah & Manan Shah. A comprehensive review on intelligent traffic management using machine learning algorithms | Innovative Infrastructure Solutions

[20] **(2022)** Comparative Study of Shortest Distance Calculation Techniques in IoT-Based Wireless Sensor Networks by Ramandeep Gill & Tarun Kumar Dubey.

Comparative Study of Shortest Distance Calculation Techniques in IoT-Based Wireless Sensor Networks | SpringerLink

[21] Google.com https://tse4.mm.bing.net/th?id=OIP.X49ZySMwVJqsCyDlPut57gAAAA&pid=Api&P=0&h=180

[22] Use of Haversine Formula in Finding Distance Between Temporary Shelter and Waste End Processing Sites by Rezania Agramanisti Azdy and Febriyanti Darnis. Use of Haversine Formula in Finding Distance Between Temporary Shelter and Waste End Processing Sites - IOPscience

[23] **(2021)** Parking Demand vs Supply: An Optimization-Based Approach at a University Campus. Parking Demand vs Supply: An Optimization-Based Approach at a University Campus - Nadimi - 2021 - Journal of Advanced Transportation - Wiley Online Library

[24] **(2023)** IoT-Enabled Smart Parking: Enhancing Efficiency and Sustainability in Smart Cities. (PDF) IoT-Enabled Smart Parking: Enhancing Efficiency and Sustainability in Smart Cities

[25] **(2023)** Revolutionizing Urban Mobility: IoT-Enhanced Autonomous Parking Solutions with Transfer Learning for Smart Cities. (PDF) Revolutionizing Urban Mobility: IoT-Enhanced Autonomous Parking Solutions with Transfer Learning for Smart Cities

[26] Wikipedia Haversine formula - Wikipedia

[27] Census of India 2011, India Infrastructure Report 2023, National Housing Bank (NHB). CSEP — India's Housing Vacancy Paradox: How rent control and weak contract enforcement produce unoccupied units and a housing shortage at the same time - CSEP csep.org,

[28] CSEP — Roundtable | Vacant housing in urban India – Extent, cause & policy solutions | Brookings brookings.edu

Rishabh Jain

PCS25-09

Quick Submit Quick Submit KIET

Group of Institutions, Ghaziabad

Document Details

Submission ID
trn:oid:::1:3240899206

Submission Date
May 6, 2025, 9:14 AM GMT+5:30

Download Date
May 6, 2025, 9:31 AM GMT+5:30

File Name
PCS25_09_Project_Report.pdf

File Size
2.7 MB

69 Pages

14,444 Words

87,632 Characters

24% detected as AI

The percentage indicates the combined amount of likely AI-generated text as well as likely AI-generated text that was also likely AI-paraphrased.

Caution: Review required.
It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Detection Groups

67AI-generated only16%
Likely AI-generated text from a large-language model.

20AI-generated text that was AI-paraphrased8%
Likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

Disclaimer
Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?
The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI-paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?
Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.