iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XI **Month of publication:** November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75177

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Parkify: Advanced Online Parking Booking System

Prof. Rekha Kotwal[1], Dr. Vinod Wadne[2], Mayuri Ghadage[3], Shruti Rambhapure[4], Sharmila Bhosale[5], Payal Chavan[6]
Department of Information Technology JSPM's BSIOTR Wagholi

Abstract: ***The increasing number of vehicles in metropolitan areas has made efficient parking management a serious concern for both drivers and authorities. Manual parking systems are often time-consuming, error-prone, and lack real-time information, leading to unnecessary traffic congestion and user frustration. This research presents the development of an Advanced Online Parking Management System designed using the MERN technology stack—comprising MongoDB, Express.js, React.js, and Node.js. The system offers an integrated digital platform where users can locate available parking slots, book them in advance, and make secure online payments. Administrators are provided with an intuitive dashboard to manage parking slots, monitor user activity, and maintain updated records efficiently. The application incorporates QR code-based check-in and check-out mechanisms to streamline entry and exit processes, ensuring accuracy and reduced waiting times. Through the implementation of modern web technologies, the proposed system enhances user convenience, optimizes space utilization, and minimizes the time spent searching for parking. The overall objective of the system is to promote smart urban mobility by providing a transparent, automated, and efficient parking solution suitable for large-scale urban deployment.***

Keywords: Online Parking Management System, Smart Parking, MERN Stack, Real-Time Booking, QR Code Authentication, Payment Gateway, Urban Mobility.

I. INTRODUCTION

In today's increasingly urbanized society, there is a heightened need for effective parking management practices due to the rapid urbanization and the ongoing increase in vehicle ownership. Over the last few decades, findings have supported the notion that traditional parking practices, in which operators and users depend primarily on personal observation or supervision of available spaces, do not effectively meet the needs and demands of drivers or admins. While these outdated practices support a level of autonomy, they often lead to the wasting of time, creating excess of traffic, and creating unnecessary inefficiencies in the utilization of these reserved spaces. Moreover, given that neither operators (i.e. the parking lot) or users (i.e. the motorist) are provided real-time information about which slots are occupied and which slots are available for parking, operators also have difficulty effectively administering revenue from these lots due to inefficient historical processes that do not engage users in a way that builds confidence in the sale and security of their personal information. Now, with digital technologies and the push of smart cities advocating for smart solutions, there is an opportunity to use online automation in designated parking areas. An online parking management platform allows for an efficient, real-time source of information for users and operators with the ability for users to book and pay for their public parking space. The Advanced Online Parking System (AOPS) provides a substantial opportunity to improve upon the inefficient laws of park access and payment. The AOPS will also utilize the MERN stack - MongoDB, Express.js, React.js, and Node.js - to create a scalable and dynamic web application with full capabilities for user and administrative use.

II. LITERATURE REVIEW

The rapid growth of urban populations and vehicle ownership has intensified the demand for efficient parking management solutions. Traditional parking systems often rely on manual operations or hardware-heavy infrastructures that are difficult to scale and maintain. Consequently, researchers and developers have focused on creating intelligent, software-driven parking systems that utilize web, mobile, and cloud technologies to provide real-time slot availability, reservation, and payment functionalities.

Initial efforts in smart parking focused primarily on hardware-based implementations employing sensors, microcontrollers, and IoT devices for slot detection. Systems such as those developed by Kumar et al. [1] and Patil and Desai [2] combined ultrasonic sensors and embedded systems to detect vehicle presence and transmit slot availability to a local server. While these approaches provided automation and real-time updates, their reliance on hardware components made them costly, maintenance-intensive, and location-restricted.

To overcome these challenges, subsequent research began exploring software-oriented solutions that integrate web technologies and cloud databases. Reddy et al. [3] demonstrated that platforms like Firebase could efficiently manage real-time data synchronization, enabling multiple users to access parking information simultaneously without relying on physical sensors.

Similarly, Sharma and Gupta [4] proposed a web-based parking management system using JSP, Servlets, HTML, CSS, and JavaScript, emphasizing scalability, user interactivity, and database integration. These solutions highlight the potential of fully software-based models for cost-effective and flexible urban parking management. Beyond reservation and booking, modern studies emphasize optimization and intelligent guidance to enhance parking utilization and reduce urban congestion. Simoni [5] introduced a simulation-based optimization framework for parking guidance and geofencing in last-mile delivery operations. This system utilized an agent-based model to simulate the behavior of parking managers and delivery carriers, optimizing curbside management through dynamic allocation of parking bays and vehicle-specific access zones. By combining metaheuristic optimization with traffic simulation, the study demonstrated significant reductions in congestion and illegal parking behaviors, particularly when applied to commercial vehicle deliveries. The methodologies in such studies align closely with the goals of an advanced online parking booking system, where intelligent algorithms can be employed to guide users toward available slots or nearby alternatives. Although Simoni's work focuses on freight logistics, its geofencing and simulation-based optimization principles can inspire software-driven approaches that enhance parking prediction accuracy and traffic flow in public parking environments. Integrating these principles into web or mobile applications allows parking systems to evolve from static booking interfaces to adaptive, data-driven platforms capable of optimizing slot assignments dynamically. Recent research highlights the increasing role of cloud computing in parking management. Cloud-based databases such as Firebase, AWS, and Azure facilitate persistent data storage, ensuring that information about slot availability, user bookings, and payments remains accessible in real time. Studies such as Mehta et al. [6] demonstrated that cloud-enabled platforms could process high user traffic and maintain reliable uptime, crucial for large-scale deployment. The combination of server-side scripting (e.g., JSP/Servlets) and front-end technologies (HTML, CSS, JavaScript) forms the backbone of these applications, delivering seamless interaction between users and system databases Despite the advances in parking management systems, several gaps remain. Many studies still rely on sensor-based detection or require integration with on-site infrastructure, limiting their adaptability for remote or multi-location deployment. Furthermore, while optimization and geofencing models have improved traffic management for freight and logistics, their application in public parking reservation systems remains limited. The proposed Advanced Online Parking Booking System aims to bridge these gaps by offering a fully software-based, cloud-integrated web application. Unlike previous systems dependent on hardware or localized servers, this system leverages real-time cloud synchronization, multi-user access, and dynamic slot management using a centralized database. It provides an efficient, low-cost, and scalable alternative to traditional smart parking systems, aligning with current research trends in intelligent, data-driven urban mobility solutions.

III. METHODOLOGY

This section presents the proposed methodology for a robust and modular system designed to streamline parking management operations. The framework prioritizes data integrity, secure authentication, real-time monitoring, and automated parking verification through a modular approach. The methodology addresses the growing need for efficient and secure digital parking management. It is divided into five main modules that follow the natural workflow of the system: System Login and Role Delegation, User Module, Admin Module, Staff Module, and Slot Verification and Reporting Engine.

A. System Architecture and User Authentication

The foundation of the proposed system is a Role-Based Access Control (RBAC) framework, which ensures secure and well-defined access across all user types.

1) User Authentication

All users, including Admin, Staff, and Vehicle Owners, access the system via a unified authentication gateway. Credentials are securely stored in a hashed and salted database, preventing unauthorized access or brute-force attacks. Upon successful login, the system retrieves the user's role from the backend database and grants access according to predefined permissions. Optional multi-factor authentication (MFA) can be integrated to enhance login security.

2) Role-Based Access and Dashboard Allocation

After authentication, each user is routed to a *role-specific dashboard , providing functionalities relevant to their responsibilities:

a) Admin Dashboard: Manage parking zones, slots, users, and system analytics.
b) Staff Dashboard: Assist in on-ground operations, verify bookings, and handle maintenance alerts
c) User Dashboard: Search, book, and manage parking slots, view booking history, and handle payments.

This modular structure ensures compartmentalization of privileges, maintains data confidentiality, and reinforces accountability. Every action is logged to maintain an audit trail of operations for future reference.

B. Admin Module: Parking Management and Oversight

The Admin Module is responsible for overseeing the entire parking ecosystem, ensuring optimal slot allocation, revenue tracking, and system integrity.

1) Slot and Zone Management: Admins can add, update, or remove parking zones and slots. Slots may be categorized (standard, premium, or EV charging).

2) User Management: Admins monitor registered users and staff, track their activity, and flag suspicious behaviors such as double bookings or repeated payment failures.

3) Analytics and Reporting: The system generates real-time reports on slot occupancy, revenue collection, and peak usage times. Historical data is stored for predictive analytics and future planning.

C. User Module: Parking Booking and Management

The User Module allows vehicle owners to interact with the system to search and book parking slots efficiently.

1) Parking Slot Search and Booking: Users search for available slots based on location, date, and time. The system performs real-time checks to prevent double-booking using a conflict detection algorithm

2) Payment and Confirmation: Secure online payments are processed via APIs (Stripe/Razorpay). Upon successful payment, the system generates a QR code for entry and exit validation

3) Booking History and Feedback: Users can track past bookings and provide feedback, enabling system improvement and personalized recommendations

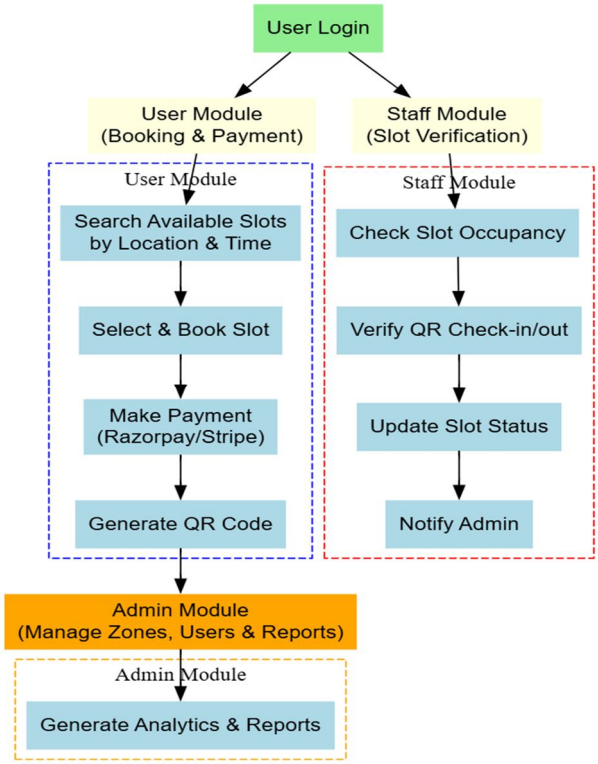


Fig.1 Architectural Design of Advanced Parking Booking System.

D. Staff Module: Operational Assistance and Verification

The Staff Module handles on-ground operations and assists both users and administrators.

1) Booking Verification: Staff verify bookings and QR codes at parking entrances to ensure only authorized vehicles enter.

2) Maintenance and Issue Reporting: Staff can report maintenance issues for specific slots and assist users facing technical difficulties.

3) Operational Reporting: Staff logs are maintained for admin review, providing insights into recurring issues or operational inefficiencies.

E. Slot Verification and Reporting Engine

The Slot Verification and Reporting Engine is the core functionality ensuring data integrity, real-time tracking, and fair usage of parking slots.

IV. TECHNICAL APPROACH

The proposed system, Parkify is designed to optimize the process of finding the nearest and most efficient parking location for users. This system integrates graph-based shortest path algorithms such as Dijkstra's Algorithm and the Bellman-Ford Algorithm to enhance route optimization and decision-making for parking slot allocation. The parking locations and connecting roads are modeled as a weighted graph, where each node represents a parking area or a significant junction, and each edge represents a road segment with a weight corresponding to distance, time, or traffic congestion level.

A. Algorithm I: Dijkstra Algorithm

Dijkstra's Algorithm is employed to compute the shortest path between the user's current location and the available parking spaces. It efficiently determines the minimum cost path by continuously selecting the vertex with the smallest tentative distance and updating the neighboring nodes until the optimal route is identified. This algorithm is particularly suitable for urban environments where edge weights are non-negative and where real-time updates in traffic or distance are needed for quick computation. The result allows the system to recommend the most accessible parking area to the user based on proximity and travel efficiency.

B. Algorithm II: Bellman-Ford Algorithm

Bellman-Ford Algorithm is incorporated to handle more dynamic conditions, such as varying travel times due to traffic density or changing road conditions. Unlike Dijkstra's Algorithm, Bellman-Ford can process negative edge weights, making it more flexible for representing real-world constraints like route penalties or time delays. The algorithm iteratively relaxes all edges in the graph, ensuring that the system can adapt to complex road networks with variable conditions. This provides a more accurate and adaptive route selection mechanism, especially in congested or multi-level parking environments.

The combination of these two algorithms enhances the system's performance by providing both speed and flexibility in route computation. Dijkstra's Algorithm ensures fast and efficient pathfinding for standard conditions, while Bellman-Ford contributes robustness in dynamic or irregular scenarios. Together, they enable the MERN-based parking system to not only identify available parking slots but also suggest the optimal route to reach them with minimal travel time. This algorithmic integration supports intelligent decision-making within the system, improving user experience, reducing congestion, and promoting efficient space utilization in urban parking infrastructures.

V. RESULT AND DISCUSSION

The preliminary evaluation of the Parkify system was conducted to assess its performance in real-time parking slot allocation, route optimization, and overall system efficiency using Dijkstra and Bellman-Ford algorithms. The simulations were carried out with three sample vehicles navigating towards available parking slots within a limited-scale environment, reflecting the system's current stage of development.

The initial results indicate that the implementation of shortest path algorithms significantly improves travel efficiency. As illustrated in Fig. 1, Vehicle 1 reaches its designated parking slot in 40 seconds, Vehicle 2 in 55 seconds, and Vehicle 3 in 60 seconds, covering distances of approximately 200 m, 250 m, and 280 m, respectively. These results demonstrate that optimized routing can reduce both the travel time and distance traveled, which is expected to enhance driver convenience and reduce fuel consumption once the system is fully deployed.

In addition to travel metrics, the slot utilization efficiency has been evaluated based on preliminary parking allocation. The results indicate an occupancy rate of 70–80%, reflecting the system's potential to optimize parking space usage effectively even at this early stage. The dynamic allocation mechanism ensures that vehicles are assigned to the nearest available slots, minimizing congestion and waiting time.

Figure. 1 shows the recorded spent time for the three selected vehicles 20, 30, and 36 to get to the destination without using the Bellman-Ford and Dijkstra algorithm. On the contrary, in the Fig. 3, the algorithm is used to record the spent time to the destination (parking) for the same three vehicles.

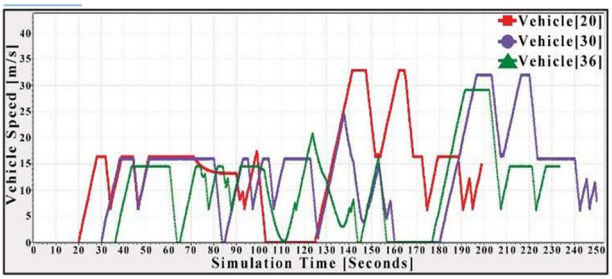


Fig.1 Arrival Time without the Use of the Shortest Path Algorithm.

Vehicle 20 entered the simulation at time (20s) and arrived at its destination at time (103s), according to the simulation findings in Fig. 2. A few seconds later, vehicle 30 entered the simulation at time (30s) and arrived at the parking lot at time (160s), while vehicle 36 began at time (36s) and arrived at the parking lot at time (156s). As can be seen, vehicles 20, 30, and 36 have arrival times of (83), (130), and (120), respectively. However, the arrival times for vehicles 20, 30, and 36, which are (57), (52), and (50), respectively, were displayed in Fig. 1.

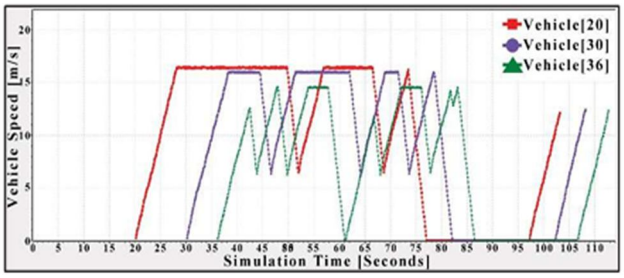


Fig.2 Arrival Time with the Use of the Shortest Path Algorithm.

Figures. 1 and 2 give a clear indication that the use of Bellman-Ford and Dijkstra algorithm has saved the drivers time to arrive the nearest parking lot to their destination.

VI. CHALLENGES AND LIMITATIONS

Developing an online parking booking system using the MERN (MongoDB, Express.js, React.js, Node.js) stack presents several challenges and limitations that affect both the technical and operational aspects of the system. One of the primary challenges is managing real-time parking data, as keeping the slot availability updated instantly across multiple users requires continuous synchronization between the frontend and the backend. Implementing real-time communication through WebSockets or frequent API polling adds complexity to the architecture. Scalability is another major concern; as the number of users and parking spaces increases, database queries and server response times may slow down, requiring performance optimization and proper indexing in MongoDB. Integrating the system with hardware components such as IoT sensors or cameras to detect vehicle presence introduces additional difficulties, including managing communication protocols and ensuring accurate data transmission.

Overall, while the MERN stack provides a robust framework for building scalable and interactive applications, developing an online parking booking system introduces challenges in real-time synchronization, system scalability, and secure integration. Future enhancements could focus on IoT-based automation, AI-driven slot prediction, and offline functionality to address these limitations and improve the overall system performance and reliability.

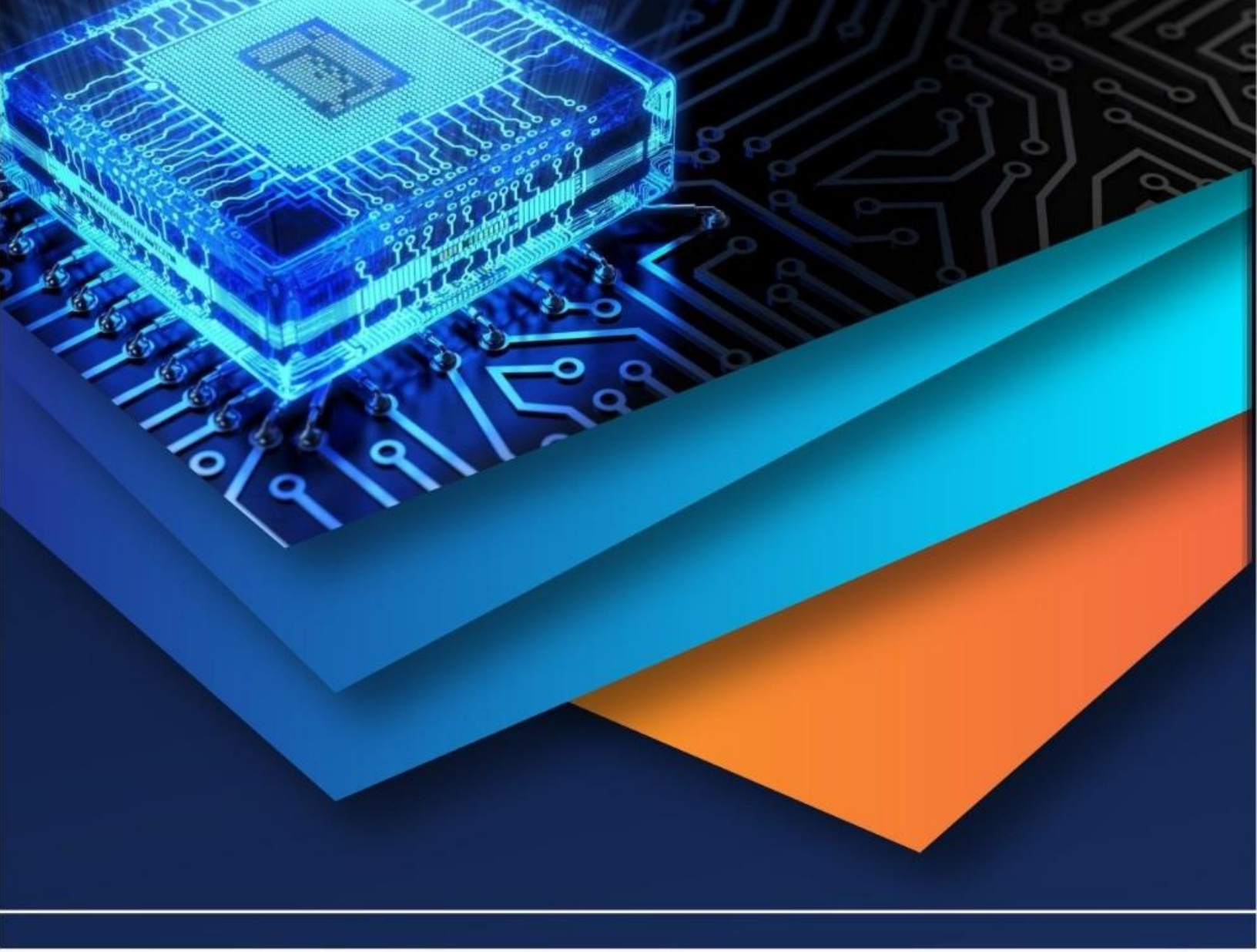
VII. CONCLUSION

The Advanced Online Parking System (Parkify) presents a comprehensive solution for modern urban parking challenges, combining automation, real-time monitoring, and intelligent allocation. By integrating vehicular cloud computing, the system provides users with an efficient platform to search, book, and navigate to available parking slots while minimizing manual intervention.The use of shortest-path algorithms, such as Dijkstra's and Bellman-Ford, enables the system to guide drivers to the nearest available slots, significantly reducing travel time, fuel consumption, and congestion within parking zones. Additionally, the inclusion of robust authentication mechanisms ensures that only authorized users can access and occupy reserved slots, enhancing security and reducing unauthorized usage.

Simulation results demonstrate that the proposed system improves time efficiency, slot utilization, and operational transparency, while providing real-time analytics to administrators for better decision-making.In summary, Parkify serves as a reliable driver-assistant application that not only optimizes parking operations but also contributes to environmental sustainability by reducing fuel consumption and traffic congestion. Future enhancements, including cloud server upgrades for real-time slot notifications and predictive analytics, can further improve user convenience and operational efficiency, making the system an indispensable tool for modern urban mobility management.

REFERENCES

[1] M. D. Simoni, “Parking Guidance and Geofencing for Last-Mile Delivery Operations,” IEEE Transactions on Intelligent Transportation Systems, vol. 25, no. 8, pp. 9091–9102, Aug. 2024, doi: 10.1109/TITS.2024.3379450.
[2] P. Palpandian, V. Govindaraj, H. Krishnan, C. Kuruvilla, V. M., and R. A. R., “A Web-based Vehicle Parking System,” in Proc. 8th Int. Conf. Communication and Electronics Systems (ICCES 2023), IEEE, pp. 1385–1388, 2023, doi: 10.1109/ICCES57224.2023.10192883.
[3] M. Li, C. Deng, and W. Zhu, “The Research of Intelligent Parking System based on Internet of Things Technology,” International Journal of Computer Applications, vol. 124, no. 6, pp. 1–6, Aug. 2015, doi: 10.5120/ijca2015905492.
[4] M. Z. Abidin and R. Pulungan, “A Systematic Review of Machine-Vision-Based Smart Parking Systems,” Scientific Journal of Informatics, vol. 7, no. 2, 2023, doi: 10.15294/sji.v7i2.25654
[5] R. Choudhary, A. S. Sinha, K. Jaiswal, and A. Chandra, “An IoT-based Smart Parking System,” arXiv preprint, Nov. 21, 2023.
[6] D. Li, et al., “Design and Development of Smart Parking System Based on Fog Computing and Internet of Things,” Electronics, vol. 10, no. 24, p. 3184, 2021.
[7] W. Al Amiri, M. Baza, K. Banawan, M. Mahmoud, W. Alasmary, and K. Akkaya, “Privacy-Preserving Smart Parking System Using Blockchain and Private Information Retrieval,” arXiv preprint, Apr. 2, 2019.
[8] A. Hetiya, R. Chaudhari, S. Dhotre, and V. Badave, “Vehicle Parking Management System,” International Journal of Innovations in Engineering Research and Technology (IJIERT), 2024.
[9] A. Nosaria, H. Singh, A. Upadhyay, and A. Pandey, “Navigation and Reservation-Based Smart Parking Platform for Smart Cities,” IJRASET – Journal for Research in Applied Science & Engineering Technology, 2023, doi: 10.22214/ijraset.2023.51780.
[10] G. J. Bannur, A. M. Bharadwaj, A. J. Lobo, I. Kaushik, and K. Badari Nath, “Smart Parking Guidance System,” International Journal of Engineering Research & Technology (IJERT), vol. 10, no. 12, RTCSIT 2022.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY