International Journal of Science, Strategic Management and Technology
Volume **02** Issue **04** April-**2026** | ISSN: **3108-1762 (Online)** | Impact Factor: 3.8
An International, Peer-Reviewed, Open Access Scholarly Journal Indexed in recognized academic databases

Blockchain-Enabled Framework for Slot Booking and Transportation Coordination in Large-Scale Events

Author Details:

Pravin R. Pachorkar[1], Vaishnavi B. Ghuge[2], Rati R. Antapurkar[3] , Vedika S. Gaikwad[4] , Vedika L. Bhusari[5] , Ankita P. Shinde[6]

[1] Assistant Professor, Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra, India
[2] UG - Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra, India
[3] UG - Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra ,India
[4] UG - Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra, India
[5] UG - Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra, India
[6] Assistant Professor, Information Technology Engineering, SPPU University, MVP's KBTCOE, Nashik, Maharashtra, India
CorrespondingAuthorEmails:pachorkar.pravin@kbtcoe.org[1],vghuge1216@gmail.com[2], ratiantapurkar001@gmail.com[3],vedikagaikwad1324@gmail.com[4], vedikabhusari22@gmail.com[5],shinde.ankita@kbtcoe.org[6]

https://doi.org/10.55041/ijsmt.v2i4.161

Cite this Article: Pachorkar, P. R., Ghuge, V. B., Antapurkar, R. R., Gaikwad, V. S., Bhusari, V. L. & Shinde, A. P. (2026). Blockchain-Enabled Framework for Slot Booking and Transportation Coordination in Large-Scale Events. International Journal of Science, Strategic Management and Technology, *02*(04). https://doi.org/10.55041/ijsmt.v2i4.161

License: This article is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting use, distribution, and reproduction in any medium, provided the original author(s) and source are properly credited.

Abstract—

Managing transportation and crowd movement during large religious gatherings is a major challenge because thousands of pilgrims travel within a short period of time. Traditional manual methods of registration and transport planning often lead to repeated registrations, overcrowding, delays, poor coordination, and difficulty in maintaining records. This paper presents a blockchain-based system for slot booking and transport management designed for large-scale religious events. The proposed system integrates village-wise registration, travel slot selection, bus assignment, and QR-code ticket generation into a single web-based platform. Blockchain technology is used to securely store booking information, ensuring that records remain transparent and cannot be altered easily.system integrates village-wise registration, travel slot selection, bus assignment, and QR-code ticket generation into a single web-based platform. Blockchain technology is used to securely store booking information, ensuring that records remain transparent and cannot be altered easily. The system also enables quick passenger verification through QR-code tickets before boarding. A web application was developed for both pilgrims and administrators. Pilgrims can register, select their preferred travel slot, and receive a digital ticket, while administrators can monitor bookings, assign buses based on passenger count, and manage the overall schedule. The implementation demonstrates that the system reduces manual effort, improves crowd management, enhances security, and

,provides a more efficient and reliable approach to transportation management during religious gatherings.

Keywords— Blockchain, Crowd Management, Transportation Management, Slot Booking, QR Code Ticketing, Bus Scheduling

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

I. INTRODUCTION

Large-scale religious gatherings in India, such as the Kumbh Mela, involve the movement of thousands of pilgrims from different villages, towns, and districts within a limited period. Managing transportation during such events is a challenging task because authorities must coordinate registrations, assign buses, maintain travel schedules, and verify passengers efficiently. Even a small error in planning can create confusion, overcrowding, delays, and inconvenience for a large number of people.At present, transportation arrangements for many religious events are still handled manually. Pilgrims often submit their details to local coordinators, who record the information in registers or notebooks.

Ticket verification is performed manually, while departure times and bus assignments are communicated separately. Such practices are time-consuming and frequently result in duplicate registrations, overcrowded buses, empty seats in some vehicles, and uncertainty among passengers regarding their assigned bus or departure slot. Since information is stored in different locations and managed by separate systems, administrators face difficulty in maintaining records and coordinating transportation effectively.Existing studies have highlighted the importance of digital systems in improving crowd and transportation management. Blockchain-based frameworks have been proposed for crowd coordination and trust management because they provide transparency, decentralization, and protection against data manipulation [1], [2]. Similarly, recent research has shown that blockchain can enhance transportation systems by improving the security, reliability, and transparency of travel records and ticketing processes [9]–[11]. QR-code-based ticket verification has also emerged as an effective method for reducing fraud and speeding up passenger validation [5], [21].Although prior studies have proposed blockchain-enabled ticketing and transportation solutions, most of them focus only on a single aspect, such as ticket issuance, payment systems, or seat allocation [5], [14], [17].

Very few studies provide an integrated platform that combines village-wise registration, travel slot booking, bus allocation, passenger verification, and schedule management for large public gatherings. Moreover, many existing systems are designed for urban transportation and assume that users possess high digital literacy and constant internet access. These assumptions are not suitable for rural pilgrims who may require a simple and mobile-friendly platform.Therefore, a significant research gap exists in the development of an integrated transportation management system specifically designed for large religious gatherings. The present work addresses this gap by implementing a blockchain-based web application that combines registration, slot booking, bus allocation, and QR-code ticket verification within a single platform. Blockchain is used to securely store booking details, making records difficult to alter and reducing the possibility of duplicate or fraudulent entries. The generated QR-code ticket enables quick and accurate passenger verification at the boarding point.

The main objectives of this research are:

1. To design a unified platform for village-wise registration, travel slot booking, and bus allocation.
2. To implement blockchain technology for secure and tamper-resistant storage of booking records.
3. To develop a QR-code-based ticket verification system for faster passenger validation.
4. To reduce manual effort, overcrowding, scheduling confusion, and

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

record-management errors during large religious events.

II. RELATED WORK

Many researchers have worked on ticket booking, passenger verification, and transport management. From the earlier studies, the work can be divided into three main approaches: blockchain-based ticketing, QR-code based verification, and transportation management systems. Each approach solves one part of the problem, but none of them provide all the required features together. [1], [17],[21],[25]

II.I Blockchain-Based Ticketing Systems

In the first approach, blockchain is used to store ticket and passenger details.[1] Once the booking is stored, it cannot be changed easily. Because of this, fake tickets and duplicate entries can be reduced.[4] Blockchain has been used in railway ticketing, public transport, and event ticket booking systems.[11] Smart contracts are also used in some systems to generate and verify tickets automatically.[25],[26] These systems improve security and transparency, but most of them are made only for railway booking or public transport in cities. They do not provide travel slot selection or bus planning for people coming from different villages to the same event.

II.II QR-Code Based Verification Systems

The second approach uses QR-code based tickets. After registration, the user receives a QR code on the mobile phone.[21],[24] This QR code is checked before entering the bus or travel area. The checking process becomes faster and there is no need to carry a paper ticket.[20] QR-code systems are useful when a large number of people need to be verified in a short time.Even though these systems make ticket checking easier, most of them only focus on verification.[26] The booking details and passenger records are usually stored separately in a normal database. Because of this, there is still a chance of repeated entries and confusion in managing the records.[17],[20]

II.III Transportation and Crowd Management Systems

The third approach is related to transport and crowd management.[15],[27] These systems help in bus scheduling, passenger counting, and managing the movement of people during large events. They help the authorities decide how many buses are required and how passengers should be arranged.[15],[17] However, most of these systems use separate records for registration, ticketing, and transport management. [25],[26] Since all information is not available in one place, the complete process becomes difficult to manage when many people are travelling together. These systems are also mainly tested in cities and controlled environments.[1],[15] From the earlier work, it can be seen that one system provides secure storage, another gives faster ticket checking, and another helps in transport planning. But no single system combines all these features together.[23],[26] In the proposed project, all three approaches are used in one platform. The system provides registration, travel slot selection, village-wise bus allocation, QR-code ticket generation, and blockchain-based storage. Because all the information is available in one place, the process becomes easier and more organized for both passengers and administrators.

III.SYSTEM OVERVIEW

The proposed GramYatra system consists of a React.js based frontend web application divided into a pilgrim booking portal and an administrative dashboard, a Node.js and Express REST API backend, a MongoDB database for scalable data storage, and a Polygon-based

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

blockchain integration module for secure data immutability. The overall operation begins when a user logs into the platform, browses the registered village routes, and selects an available travel or darshan slot for the Kumbh event. The system then guides the user through a booking flow where they input primary passenger details, add optional members, and securely complete the reservation via a mock payment gateway. Upon selecting a journey, the system captures user details, including Aadhar numbers and age. A specialized Quota Logic automatically categorizes passengers according to Gender, Normal, Senior Citizen, or Disabled groups. This ensures that the system adheres to social welfare policies. Validation is performed both on the frontend and via Express middleware to ensure data sanitization. Upon payment, the Node.js backend processes the data and saves the booking records to the MongoDB database. For enhanced security, a background cron job runs periodically to compute a cryptographic hash of the booking details and commits it to a custom smart contract on the blockchain using Ethers.js. The booking data is passed through a SHA-256 hashing algorithm. This creates a unique digital fingerprint of the ticket while maintaining passenger privacy. The server uses Ethers.js to sign a transaction with the administrator's private key. This transaction calls the function of the deployed Solidity contract. The booking hash and its timestamp are permanently stored on the Polygon blockchain. In parallel, administrators can use the admin dashboard to manage village profiles, adjust bus maintenance statuses, create new slots, and monitor real-time booking analytics. The final output is a confirmed digital booking ticket for the user which includes a verifiable blockchain transaction link proving the data's integrity along with a continuously synchronized administrative dashboard reflecting the latest system capacities, active bus availability, and verified bookings.

III.I Components Used

To build GramYatra, the project utilizes a modern full-stack architecture integrated with decentralized storage to ensure a seamless and secure booking experience for rural tourism.

Frontend Application Module: Developed using React.js and CSS, this module provides a responsive and intuitive interface. It allows pilgrims and tourists to browse rural attractions, view real-time bus availability, and manage their travel itineraries. It also includes a dedicated administrative dashboard for system managers to update village data, monitor bus logistics, and oversee the entire transport ecosystem.

Backend REST API Server: Built using Node.js and Express.js, this component acts as the central orchestrator. It processes incoming HTTP requests, validates user inputs, and executes the core business logic for reservations. It also ensures secure communication between the frontend and the database.

Database Component: MongoDB Atlas is used as a cloud-based NoSQL database to store all dynamic application data. It manages user profiles, village details, bus schedules, and historical reservation records. Its document-based structure allows high scalability and flexible data updates.

Blockchain Storage Module: This module uses Solidity-based smart contracts deployed on a blockchain network. Instead of storing full data, it records immutable cryptographic hashes of each booking. This ensures tamper-proof storage, transparent verification, and prevents unauthorized modification of reservation records.

Figure 1. System Architecture Components

III.II Software Used

The development of GramYatra utilizes a specialized technology stack designed for scalability and data integrity.

React.js & CSS: Used to build a responsive frontend, ensuring the user interface remains performant even under heavy traffic during peak pilgrimage seasons.

Node.js & Express: Provides the asynchronous

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

backend environment necessary for handling concurrent booking requests and managing RESTful API endpoints.

MongoDB Atlas (Mongoose): A cloud-based NoSQL database used for flexible data modelling of complex rural tourism entities, such as village details and dynamic bus schedules.

Solidity & Polygon Network: The core of the decentralized layer; Solidity is used to write smart contracts, while the Polygon network provides a high-speed, low-cost environment for recording immutable booking hashes.

Ethers.js: Acts as the bridge between the backend and the blockchain, allowing the Node.js server to sign transactions and interact with deployed smart contracts.

Node-Cron: An automated task scheduler used to trigger periodic blockchain commits, ensuring that offline booking data is synced to the ledger at regular intervals.

IV. METHODOLOGY

The project followed a structured Software Development Life Cycle (SDLC) specifically tailored for the high-stakes environment of the Kumbh event.

Planning and Requirement Analysis: This phase involved identifying the unique challenges of rural transit, such as sudden surges in pilgrim numbers. We defined functional requirements for ticket booking, route mapping, and the necessity of a ”source of truth” via blockchain to prevent ticket duplication or fraud.

System Design: We designed a Modular Architecture where the frontend components are decoupled from the backend logic. This included mapping out RESTful API endpoints (e.g., /api/bookings, /api/villages) and designing the schema for the MongoDB collection to optimize query speeds for real-time bus tracking.

Component Selection And Software Setup: The environment was initialized by configuring the MongoDB Atlas cluster with appropriate IP whitelisting. Simultaneously, the contract was developed, compiled, and deployed to the Polygon Amoy testnet, with the contract address and ABI integrated into the backend environment variables.

Implementation / Development: The core logic was built in parallel: the frontend team developed the booking flows and admin dashboards using React, while the backend team implemented the cryptographic hashing functions. These functions take booking details, generate a unique hash, and prepare them for blockchain commitment.

V. PROJECT IMPLEMENTATION

The implementation of the GramYatra system progressed through several iterative stages, beginning with a detailed ground survey conducted across rural villages where direct bus transportation is not available. By interacting directly with villagers and Gram Panchayat officials, the team gathered first-hand insights into the difficulties experienced during large religious gatherings such as the Kumbh Mela. Common issues reported during the survey included the absence of organized transportation, irregular bus timings, crowd build-up on popular bathing days, lack of a digital mechanism to manage slot bookings, frequent ticket duplication, and no formal process for handling lost-and-found cases—challenges also observed in prior studies on large-scale crowd systems [1], [2]. These observations helped shape the functional requirements of the system and ensured that the solution aligned with the needs and constraints of rural communities.

Once the requirements were clearly defined, the system design and module planning phase was initiated. The platform was structured as a multi-component solution comprising a user panel for villagers, an admin panel for local authorities, and a backend capable of managing data operations while also integrating blockchain for secure verification. Web-based ticketing and management platforms proposed in earlier works [17], [19] also guided the interface choices. The user interface was developed using React.js to provide a simple and intuitive layout suitable even for first-time digital users. The admin side

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

includes dashboards showing connected villages, total bookings, and bus availability, along with tools for adding villages, managing buses, and

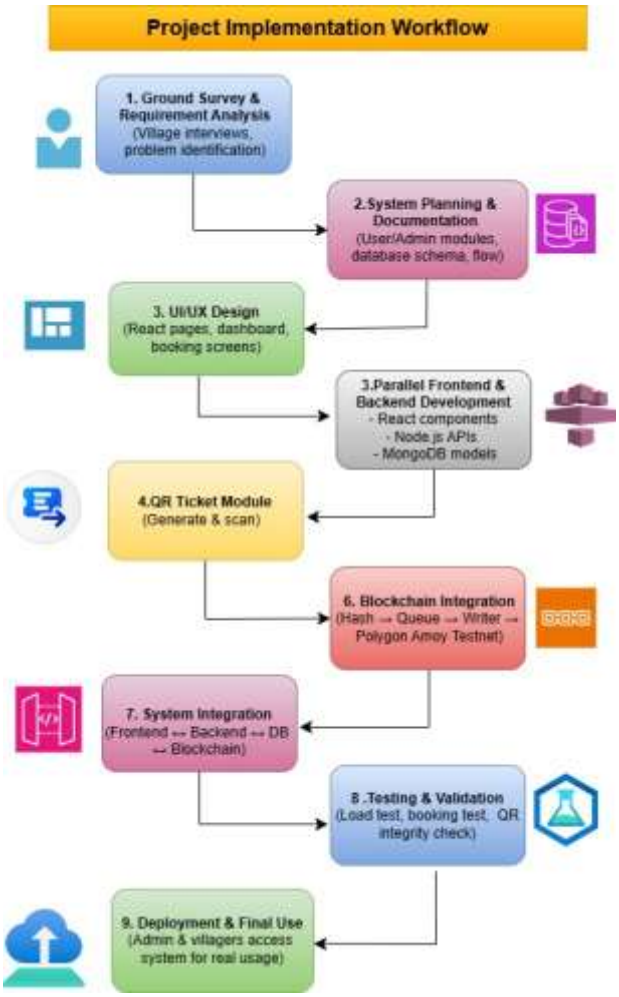

figure 2. Project Implementation Workflow

confirming bookings. Both the frontend and backend were implemented in parallel to maintain uniformity across modules and enable faster integration, following agile-like iterative approaches commonly used in transportation systems development [20].

The backend development relied on Node.js with Express.js to implement core functionalities such as user registration, slot allocation, bus assignment, and generation of digital tickets. After a user successfully books a slot, the system

generates a unique ticket ID, compiles the booking information, and produces a QR code for authentication at boarding points, similar to QR-based smart ticketing mechanisms discussed in prior literature [21]. To prevent duplication and ensure that each ticket is tamper-proof, the booking data is hashed using SHA-256. Instead of writing this data directly to the blockchain, the hash is temporarily placed in a commit queue. This intermediate step allows blockchain transactions to occur asynchronously, aligning with scalable architectures used in decentralized event ticketing frameworks [24], [25].

A specialized writer service was designed to handle blockchain commitments. This service reads pending items from the commit queue and interacts with the Polygon Amoy Testnet through Alchemy RPC services. It periodically invokes the smart contract deployed on the test network and stores the corresponding ticket hash on the blockchain. Similar blockchain-secured ticketing and verification flows have been reported in recent studies [4], [5], [16], [22], [23]. During passenger boarding, the QR code is scanned, the system recomputes the hash locally, and compares it with the one stored on the blockchain. A match verifies the ticket's authenticity, while a mismatch signals an attempt at alteration or misuse. This combined use of QR codes and blockchain validation establishes a secure and reliable ticketing mechanism capable of handling large crowds, echoing the security principles outlined in blockchain-enabled transportation systems [10], [11], [14].

MongoDB served as the database for the system, with collections dedicated to users, bookings, buses, admins, and queued blockchain entries. To improve performance, indexes were added to frequently queried fields such as date, village, ticket ID, and bus ID, similar to indexing strategies used in intelligent transport and mobility platforms [9], [15]. Alongside this, the

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

bus management module was implemented to help administrators register new buses, assign or modify routes, monitor vehicle status (active, failed, or under maintenance), and coordinate backup buses when required. This feature directly addresses one of the most common issues identified during the survey—unexpected bus breakdowns and lack of alternative transport—an operational challenge frequently highlighted in public transit research [18].

During the integration stage, the frontend, backend, database, and blockchain components were combined to function as a cohesive system. The user interface supports actions such as registration, date selection, slot booking, and ticket download, while the admin interface provides real-time village and booking insights. The system was tested with dummy records and subjected to simulated peak loads to evaluate its performance under conditions similar to actual event days. Some of the challenges faced during implementation included unstable internet connectivity in rural areas, delays caused by blockchain confirmation times, and synchronization issues during QR verification. These were addressed by adding retry mechanisms, using a queue-based commit design, and optimizing the smart contract to minimize execution overhead—an approach consistent with best practices highlighted in decentralized mobility studies [12], [13], [26],
[27].

In its final form, the GramYatra system offers a dependable digital solution for managing village-wise registrations, distributing crowds through slot booking, allocating transportation resources, and ensuring ticket authenticity using blockchain technology. By bringing together conventional web technologies and decentralized verification, the system enhances transparency, reduces the chances of fraud, and improves the overall coordination required for managing large-scale religious events, aligning with modern blockchain-driven travel and event management solutions [7], [8].

VI. RESULTS AND DISCUSSION

The performance of the implemented GramYatra system was evaluated through functional testing and simulation-based analysis. The results are presented using structured tables and figures to clearly demonstrate system efficiency, reliability, and usability.

VI.I Functional Testing Results

The system was tested for all major modules including registration, slot booking, QR-code generation, blockchain storage, and bus allocation. In addition, validation testing was performed for user authentication, duplicate prevention, and input handling to ensure data accuracy and security. The system was also evaluated under concurrent user conditions to verify its ability to handle multiple bookings without conflicts. QR-code scanning and verification were tested for quick and accurate passenger validation, while blockchain records were checked to ensure data immutability. Furthermore, bus allocation logic was validated to prevent overbooking and ensure proper passenger distribution. The system demonstrated stable performance and reliability under simulated peak load conditions.

Table I: Functional Testing Summary

Module	Expected Outcome	Result Observed	Status
User Registration	Unique user entries	No duplicate entries found	Passed

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

Slot Booking	Correct slot allocation	Slots assigned accurately	Passed
QR Code Generation	Unique ticket generation	QR generated successfully	Passed
Blockchain Storage	Secure and immutable records	Data stored without tampering	Passed
Bus Allocation	Assign bus based on bookings	Efficient allocation observed	Passed

Interpretation:

As shown in **Table I**, all system modules performed as expected. The system successfully eliminated duplicate registrations and ensured proper slot allocation. QR-code generation and blockchain storage worked reliably, confirming system correctness and security.

VI.II. System Performance Analysis

The system performance was evaluated under simulated multiple-user conditions.

Table II: Performance Evaluation

Parameter	Observation
Average Response Time	1.5 – 2.5 seconds
Concurrent Users Supported	100+ users (simulated)
Booking Processing Time	< 3 seconds
QR Verification Time	< 2 seconds
Blockchain Transaction Delay	5 – 10 seconds (asynchronous)

Interpretation:

From **Table II**, the system demonstrates fast response times for booking and verification processes. Although blockchain transactions introduce slight delays, the use of asynchronous processing ensures that the user experience remains smooth.

VI.III. User Interface and Workflow Validation

Figure 3: User Interface of Booking System

Figure 4: Village Selection Interface

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

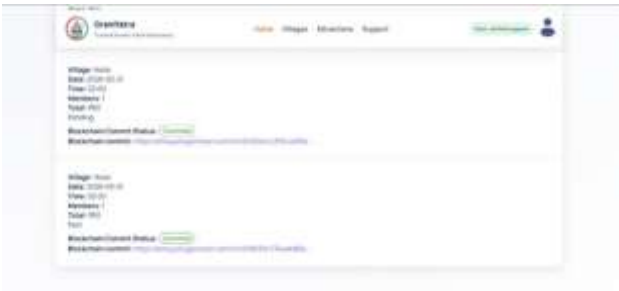

Figure 5: Blockchain-Based Ticket Booking

Interpretation:
The figures illustrate that the system provides a simple and user-friendly interface. Users can easily register, select villages, book slots, and download QR-based tickets. The admin interface enables efficient monitoring and control of bookings and transportation.

VI.IV. Comparative Analysis with Existing Systems

Table III: Comparison with Traditional and Existing Systems

Feature	**Manual System**	**Existing Systems**	**Propose d System**
Centralized Platform	No	Partial	Yes
Duplicate Prevention	No	Limited	Yes
QR Code Verification	No	Yes	Yes
Blockchain Security	No	No	Yes
Bus Scheduling Integration	No	Partial	Yes

Interpretation:
As shown in **Table III**, the proposed system integrates features that are either missing or partially implemented in existing systems. Unlike manual methods, it provides a unified platform with enhanced security and automation. Compared to earlier studies discussed in the literature, this system combines blockchain, QR verification, and transport management into a single solution.

VI.V. Key Findings

- The system significantly reduces manual errors and duplication.
- QR-based verification improves boarding efficiency.
- Blockchain integration ensures data integrity and transparency.
- The system handles multiple users efficiently under simulated load.

VI.VI. Limitations and Discussion

Despite the advantages, some limitations were observed:

- Dependence on internet connectivity, especially in rural areas
- Blockchain transaction delays due to network confirmation time
- Limited validation in real-world large-scale deployment

These limitations are consistent with challenges identified in previous studies on blockchain-based transportation and ticketing systems.

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

VI.VII Overall Discussion

The results confirm that the proposed system provides a more organized, secure, and efficient solution compared to traditional methods. Unlike earlier approaches that focused only on ticketing or verification, this system successfully integrates registration, slot booking, transport management, and secure verification into a single platform.

The use of structured tables and figures improves clarity and helps in understanding system performance and benefits effectively.

VII. CONCLUSION

This paper presented the implementation of a blockchain based slot booking and transportation management system for large-scale religious gatherings such as the Kumbh Mela. The project was successfully designed, developed, and tested using a web-based application. The results indicate that the system can effectively manage pilgrim registration, travel slot booking, bus scheduling, and secure ticket verification using QR codes and blockchain technology. The integration of these components improves transparency, security, and efficiency while reducing manual effort and operational errors. Therefore, the proposed system is suitable for managing transportation and crowd movement in large-scale religious events and can be extended to other public gathering scenarios.

VIII. FUTURE WORK

Future improvements to this project may include:

- Integration of real-time GPS tracking for buses to enhance monitoring and navigation.
- Implementation of AI-based crowd prediction and traffic management.
- Development of a mobile application for easier accessibility to rural users.
- Enhancement of blockchain scalability using Layer-2 solutions.

These enhancements can improve the system in terms of accuracy, automation, scalability, user convenience, and overall system reliability.

ACKNOWLEDGMENT

The authors would like to thank the Department of Information Technology, MVPS's KBT College of Engineering, Nashik, for their continuous support and guidance throughout the development of this project. We also express our gratitude to our mentors and faculty members for their valuable suggestions and encouragement.

REFERENCES

[1]M. H. Hasan, Y. Lee, and K. Kim, "CrowdBC: A blockchain-based decentralized framework for crowdsourcing," IEEE Transactions on Services Computing, vol. 12, no. 5, pp. 790–801, 2019.

[2]"Blockchain-based crowd-sensing trust management mechanism for crowd evacuation," in Proc. CSCWD, 2022.

[3]J. Jabez, S. Gowri, S. Vigneshwari, J. A. Mayan, and S. Srinivasulu, "Anomaly detection using CFS subset and neural network with WEKA tools," in Proc. ICTIS, 2018, pp. 675–682.

[4]P. Cholke, Y. Munde, M. Sayyed, A. Vidhale, and N. Zanwar, "Secure event ticketing system using NFT ERC721 tokens," 2024.

[5]M. Farhan, A. H. Nur, R. B. Sulaiman, U. Butt, and S. Zaman, "Blockchain-based railway ticketing: Enhancing security and efficiency with QR codes and smart contracts," IEEE Access, 2025.

[6]"Smart contract designs on blockchain applications," in Proc. ISCC, 2020.

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.

[7]"Smart travel with blockchain-based solutions," in Springer CCIS, 2024.

[8]"Blockchain in tourism: Enhancing security, reducing fraud, and revolutionizing customer engagement," in Springer CCIS, 2023.

[9]"Intelligent transportation systems based on blockchain smart contract technology in supply chain management," IEEE Xplore, 2023.

[10] "Blockchain for transportation ecosystems: Trust, security, and scalability," IEEE Xplore, 2023.

[11] Y. Chen, H. Xu, and X. Zhang, "Blockchain in public transport: Enhancing efficiency and transparency," IEEE Access, vol. 11, pp. 34567–34579, 2023.

[12] R. Singh and P. Kumar, "Blockchain-enabled mobile payment systems for smart cities," in Springer LNCS, 2023.

[13] "Decentralizing ride-sharing: A blockchain-based application with smart contract automation and performance analysis," Multimedia Tools and Applications, 2024.

[14] "Smart contracts for mobility: Decentralized ticketing and payments," in Springer CCIS, 2024.

[15] S. Banerjee, D. Das, P. Chatterjee, and U. Ghosh, "Blockchain-enabled digital twin technology for next-generation transportation systems," in IEEE ISORC, 2023, pp. 1–8.

[16] Y. YuanJiang and J. T. Zhou, "Ticketing system based on NFT," in IEEE MMSP, 2022.

[17] A. Beevi, M. Sarath, A. Krishna, M. Vahid, and A. Kumar, "Realtime bus ticket booking software for Android systems," 2024.

[18] J. F. Moore, A. Carvalho, G. A. Davis, Y. Abulhassan, and F. M. Megahed, "Seat assignments with physical distancing in public transit," IEEE Access, vol. 9, pp. 42985–42993, 2021.

[19] D. G. Chandra, R. Prakash, and S. Lamdharia, "Mobile ticketing system for automatic fare collection," in IEEE Conference, 2013.

[20] "The future of travel in public bus service," in Proc. ICAISC, 2023.

[21] "Smart ticketing for modern travel with QR code features," in Proc. ICBDS, 2024.

[22] "Applying smart contract in blockchain for ticket issuance and traceability," IEEE Xplore, 2023.

[23] "Blockchain-based intelligent ticketing system for events," IEEE Xplore, 2024.

[24] "EventOrchestr8: A blockchain-based secure QR code ticketing system," in Springer CCIS, 2024.

[25] "Blockchain-enabled e-ticketing systems: Opportunities and challenges," in Springer CCIS, 2024.

[26] "Blockchain-based ticketing for public travel: Trends and challenges," IEEE Xplore, 2024.

[27] Guo, J. Li, M. D. Nardo, and Z. Huang, "Blockchain and metaheuristic optimization-based intelligent transportation system," in ISCAIT, 2025.

© Author(s). This work is peer-reviewed, openly published, and permanently archived
This article is openly accessible and reusable with proper attribution.