Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

Event Ticket Booking with QR Code Validation Using Spring Boot and RESTful Architecture

Seru Ravi, Nittala Harshit Sai Sriram, Sundaraneedi Surya Sai Praveen, Pappula Veeranna,
Ch Siva Koppeswararao
Department of Computer Science and Engineering, Bonam Venkata Chalamayya Engineering College, Affiliated to JNTU Kakinada, Andhra Pradesh–533210, India

T. V. L. Srinivasa Rao
Project Guide, Assistant Professor, Department of Computer Science and Engineering,
Bonam Venkata Chalamayya Engineering College,
Affiliated to JNTU Kakinada, Andhra Pradesh–533210, India

Abstract - Event Ticket Booking with QR Code Validation is a web-based application designed to provide a secure, efficient, and user- friendly solution for managing event ticketing processes online. The system enables users to browse available events, book tickets, and receive digitally generated tickets in the form of unique QR codes for entry verification. By automating ticket booking and validation, the application reduces manual effort and minimizes errors associated with traditional ticketing methods.

The system is developed using modern full-stack web technologies, with React.js used for the frontend to deliver a responsive user interface, Spring Boot used for backend services to handle business logic, and MySQL used for reliable data storage and management. The application is structured into two primary modules: User Module and Admin Module. The User Module allows users to register, authenticate, view events, book tickets, and access their booked tickets through QR codes. The Admin Module enables administrators to manage events, monitor ticket bookings, and validate tickets by scanning QR codes at event entry points.

Each ticket generated by the system contains a unique QR code to ensure secure and accurate verification. Once scanned, the system validates the ticket and updates its status to prevent duplicate usage, thereby reducing ticket fraud. The application communicates through RESTful APIs to support real-time booking and validation while maintaining secure authentication and role- based access control. Overall, this project demonstrates an effective full-stack approach for building a scalable, secure, and reliable digital event ticketing system suitable for real-world applications.

Keywords **- Event Ticket Booking System, QR Code Validation, Web Application, Online Ticketing, Secure Authentication**

1. INTRODUCTION

In the digital era, event management and ticket booking have increasingly shifted toward online platforms to overcome the limitations of traditional manual ticketing systems. Conventional approaches often suffer from long queues, ticket duplication, poor record management, and lack of secure validation. This paper presents a web-based Event Ticket Booking System with QR Code Validation designed to provide a secure , efficient, and user-friendly solution for online ticket booking and event management.

The proposed system enables users to register, authenticate, browse events, book tickets, and receive digital tickets embedded with unique QR codes. Event administrators are responsible for managing events, monitoring ticket bookings, and validating tickets at entry points through QR code scanning. Each scanned QR code is verified against the database to ensure authenticity and prevent multiple usages of the same ticket.

The system is implemented using React for the frontend, Spring Boot for backend services, and MySQL for database management. Role-based access control ensures secure separation between user and administrator functionalities. By automating ticket booking and validation, the system reduces manual effort, enhances security, improves user convenience, and supports efficient event and crowd management. This project demonstrates the effective use of modern web technologies to deliver a scalable and reliable digital event ticketing solution.

The proposed Event Ticket Booking System is designed with a modular architecture to ensure scalability, maintainability, and ease of future enhancements. The separation of frontend and backend components enables independent development and efficient system updates. RESTful APIs are used for communication between the client and server, ensuring fast data exchange and real-time updates during ticket booking and validation. This architectural approach allows the system to handle multiple users simultaneously without performance degradation.

A major strength of the system lies in its QR code–based ticket validation mechanism. Each ticket is associated with a uniquely generated QR code that is securely stored in the database. During event entry, the QR code is scanned and validated in real time to confirm ticket authenticity and usage status. Once validated, the ticket status is updated to prevent reuse, thereby eliminating ticket duplication and unauthorized entry. This automated verification process significantly reduces entry delays and enhances security at event venues.

Furthermore, the system benefits event organizers by providing a centralized platform for event and ticket management. Administrators can track ticket sales, monitor attendance, and manage event details efficiently through the dashboard. The collected data can be utilized for analytical purposes, helping organizers understand user engagement and improve planning for future events.Overall, the system not only enhances user convenience but also supports organizers with reliable tools for efficient and secure event management.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

The system is designed to support future enhancements such as online payment integration, seat selection, and notification services. Its flexible architecture enables easy customization and ensures adaptability to different event types and organizational requirements.

2. LITERATURE REVIEW

Several studies have explored online event ticket booking systems to reduce manual effort and improve user convenience. Existing solutions mainly focus on web-based booking and basic ticket generation. However, many systems lack secure validation, real- time verification, and role-based access control. Recent research highlights the effectiveness of QR code–based ticket validation in preventing ticket duplication and enhancing entry security.

2.1 Web-Based Implementations

Web-based implementations of event ticket booking systems have gained significant attention due to their accessibility and ease of use. These systems allow users to browse events, book tickets, and manage reservations through web browsers without installing additional software. Most implementations use client–server architectures with centralized databases to store event and ticket information. However, many existing systems provide limited security, lack real-time ticket validation, and offer minimal support for scalability and role-based access control, which restricts their effectiveness in large-scale events.

To address these limitations, modern web-based systems increasingly adopt responsive interfaces, RESTful APIs, and secure authentication mechanisms, enabling efficient ticket management, real-time updates, and improved user experience across different devices.

2.2 Platform-Specific Solutions

Platform-specific ticketing solutions are designed for particular devices or environments such as mobile applications or proprietary systems. While they offer optimized performance and better user experience, they often suffer from limited portability, higher development costs, and restricted cross-platform compatibility.

2.3 Framework-Based Implementations

Framework-based implementations leverage established web frameworks to develop reliable and scalable event ticket booking systems. Technologies such as Spring Boot and React provide structured architectures, reusable components, and built-in security features. These frameworks support RESTful communication, efficient database integration, and modular development. Compared to traditional approaches, framework-based solutions improve maintainability, enhance performance, and simplify the implementation of features such as role-based access control and QR code–based ticket validation.

Additionally, framework-based systems enable faster development cycles, easier testing, and seamless integration of future enhancements such as online payments, notification services, and analytics, making them suitable for large-scale and real-world event management applications.

2.4 Identified Gaps

Despite the availability of various ticket booking solutions, several gaps remain in existing systems. Many lack secure real-time ticket validation, proper role-based access control, and scalable architectures. Limited integration between booking and entry verification often leads to ticket misuse. Additionally, inadequate database design and weak transaction handling reduce system reliability during high user traffic and large-scale events.

Another significant gap is the absence of responsive and user-friendly interfaces that work consistently across devices. Many systems also fail to provide analytical insights for organizers, such as attendance tracking and sales trends, limiting data-driven decision-making. Furthermore, support for future extensibility, including payment gateways and notification services, is often insufficient.

Existing solutions also show limited support for real-time synchronization between booking, validation, and administrative modules. Delayed status updates can cause inconsistencies in ticket usage records. Moreover, insufficient security measures and lack of standardized API design restrict interoperability with external systems and reduce overall system robustness.

3. SYSTEM ARCHITECTURE

The system follows a layered client–server architecture integrating frontend, backend, and database components to ensure scalability, security, and efficient ticket management.

3.1 Three-Tier Architecture

Presentation Tier:
The presentation tier provides a user-friendly and responsive interface for users and administrators. It handles event browsing, ticket booking, QR code display, and validation interactions through modern web technologies and client-side components.

Application Tier:
The application tier manages business logic, request processing, and system workflows. It implements event management, ticket booking, QR code generation, validation rules, and role-based access control through RESTful APIs.

Data Tier:
The data tier is responsible for secure storage and management of system data. It maintains records related to users, events, tickets, and QR codes, ensuring data consistency, integrity, and reliable access during system operations.

The three-tier architecture ensures clear separation of concerns, improves system scalability, simplifies maintenance, and enables efficient communication between user interfaces, business logic, and data storage layers.

3.2 RESTful API Design

RESTful APIs enable secure communication between frontend and backend, supporting real-time ticket booking, QR code validation, and role-based operations using standard HTTP methods.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

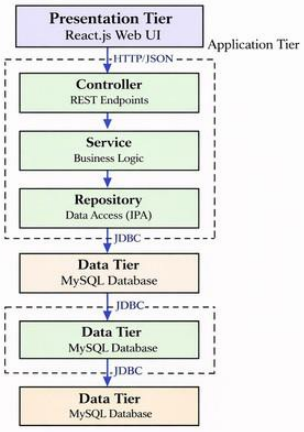

Figure 1: Three-tier architecture illustrating separation of concerns between the presentation layer (React.js Web UI), the application logic (Spring Boot), and the data layer (MySQL Database) for an event ticket booking system with QR code validation.

The figure illustrates a three-tier architecture designed for an Event Ticket Booking System with QR code validation. The Presentation Tier consists of a React.js–based web interface that allows users and administrators to interact with the system through browsers. Requests from the UI are transmitted to the Application Tier using HTTP/JSON. The Application Tier, implemented using Spring Boot, follows a layered structure comprising Controller, Service, and Repository components. Controllers expose REST endpoints, services handle business logic such as ticket booking and validation, and repositories manage data access using JPA. The Data Tier uses a MySQL database accessed through JDBC, ensuring reliable storage, data integrity, and efficient transaction handling.

3.3 Data Model and Entity Relationships

The data model of the Event Ticket Booking System is designed to support efficient storage, retrieval, and management of event- related information. Core entities include User, Event, Ticket, and QR Code. The User entity stores authentication details and role information to distinguish between users and administrators. The Event entity maintains event details such as name, date, location, and ticket availability. Each Ticket entity is associated with a specific user and event, representing a booking record. The QR Code entity contains a unique code linked to each ticket for validation purposes. Relationships between entities are defined using primary and foreign keys to ensure referential integrity. This structured data model enables accurate ticket tracking, secure validation, and efficient reporting.

Additionally, the data model includes supporting entities such as Booking, Payment, and Attendance to enhance system functionality. The Booking entity records transaction details, while Payment maintains payment status and method information. Attendance tracks ticket usage during entry validation. These relationships enable comprehensive event monitoring, prevent duplicate ticket usage, and support future system extensions.

The entity relationships are designed to minimize data redundancy and support scalability. Normalization techniques are applied to maintain consistency across tables, while indexing improves query performance.

Controller	Endpoint	Method	Purpose
Auth	/api/auth/register	POST	User registration
Auth	/api/auth/login	POST	User login
User	/api/users/{id}	GET	Get user details
Event	/api/events	POST	Create new event (Admin)
Event	/api/events	GET	List all events
Event	/api/events/{id}	GET	Get event details
Event	/api/events/{id}	PUT	Update event (Admin)
Event	/api/events/{id}	DELETE	Delete event (Admin)
Ticket	/api/tickets/book	POST	Book ticket
Ticket	/api/tickets/user/{id}	GET	Get user tickets
Ticket	/api/tickets/{id}	GET	Get ticket details
QR Validation	/api/qr/validate	POST	Validate QR ticket
	/api/qr/scanLogs	GET	Get scan logs (Admin)
Payment	/api/payments/create	POST	Process payment
Payment	/api/payments/{id}	POST	Get payment status
Payment	/api/payments/{id}	GET	Get payment status

Figure 2: RESTful endpoint organization across backend controllers.

3.4 Technology Stack Selection

The frontend of the system is developed using React.js to provide a dynamic, responsive, and user-friendly interface. React's component-based architecture simplifies UI development, enhances reusability, and ensures smooth user interaction across different devices and screen sizes.

The backend is implemented using Spring Boot, which offers a robust framework for building RESTful web services. It simplifies application configuration, supports modular development, and provides built-in features for security, validation, and transaction management, making it suitable for enterprise-level applications.

MySQL is used as the database management system to store user, event, ticket, and QR code data. It provides reliable data storage, supports complex queries, and ensures data consistency. The combination of these technologies results in a scalable, secure, and maintainable event ticket booking system.

4. SYSTEM DESIGN AND IMPLEMENTATION

The Event Ticket Booking System is designed using a modular and layered approach to ensure maintainability and scalability. The frontend communicates with the backend through RESTful APIs, while the backend processes requests using well-defined service and repository layers. Core functionalities such as user authentication, event management, ticket booking, and QR code validation are implemented with proper business rule enforcement. The system ensures secure data handling, real-time updates, and efficient transaction processing, providing a reliable and user-friendly event management solution.

4.1 Authentication and Role Management

The system implements secure user authentication with role- based access control, ensuring authorized access to user and administrator functionalities while protecting sensitive operations and system data.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

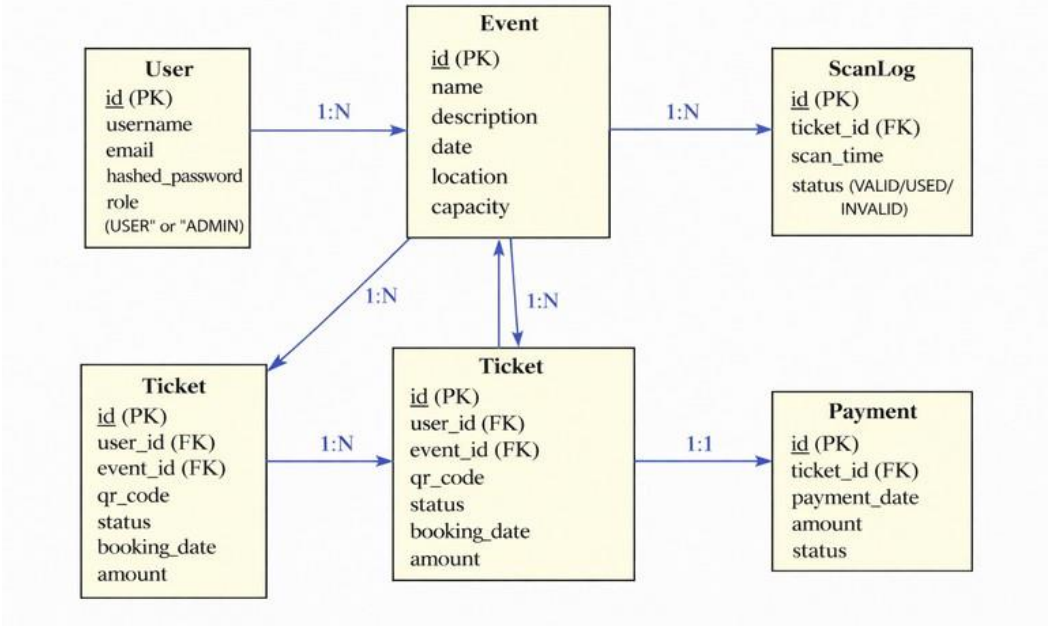

Figure 3: Entity-relationship diagram representing the normalized database schema and inter-entity dependencies.

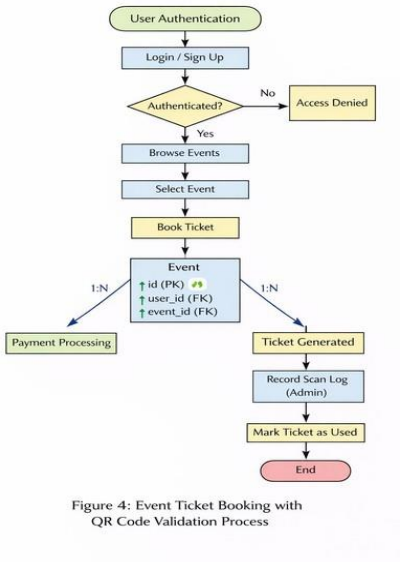

Figure 4: Event Ticket Booking with QR Code Validation Process

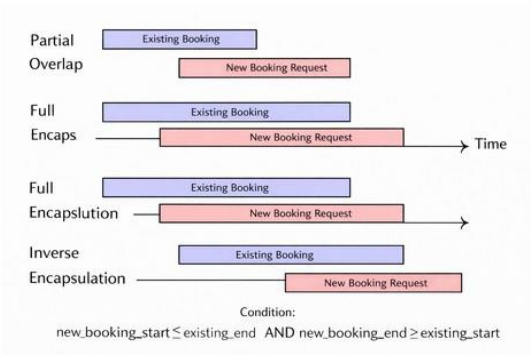

Figure 5: Illustration of time range overlap scenarios used to validate new event ticket booking requests against existing bookings and prevent scheduling conflicts.

4.2 Ticket Booking Lifecycle and Validation

The ticket booking lifecycle includes event selection, ticket reservation, QR code generation, and entry validation. Business rules ensure ticket availability, prevent duplicate bookings, and validate QR codes in real time to avoid misuse.

4.2.1 Comprehensive Validation Framework

The validation framework enforces business rules by verifying event availability, ticket limits, QR code uniqueness, and ticket status, ensuring secure booking, accurate data handling, and prevention of duplicate ticket usage.

depicts time-range overlap scenarios used to validate new event booking requests against existing bookings, ensuring conflict detection, accurate scheduling, and prevention of overlapping reservations.

4.2.2 Overlapping Booking Detection

Overlapping booking detection executes a database query to identify conflicts with existing bookings for the same event and time slot. The repository uses a JPQL query to check bookings with active statuses and intersecting time ranges:
SELECT b FROM Booking b WHERE b.event.id = :eventId AND b.status IN ('CONFIRMED', 'PENDING') AND (b.startTime <=
:endTime AND b.endTime >= :startTime)
This query retrieves bookings that satisfy three conditions: same event, non-terminal booking status, and time-range intersection evaluated using a boolean predicate. The predicate captures all overlap scenarios, including partial overlap, full encapsulation, and inverse encapsulation. If the query returns any results, the booking request is rejected with a descriptive validation error. The overlap detection logic is illustrated

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

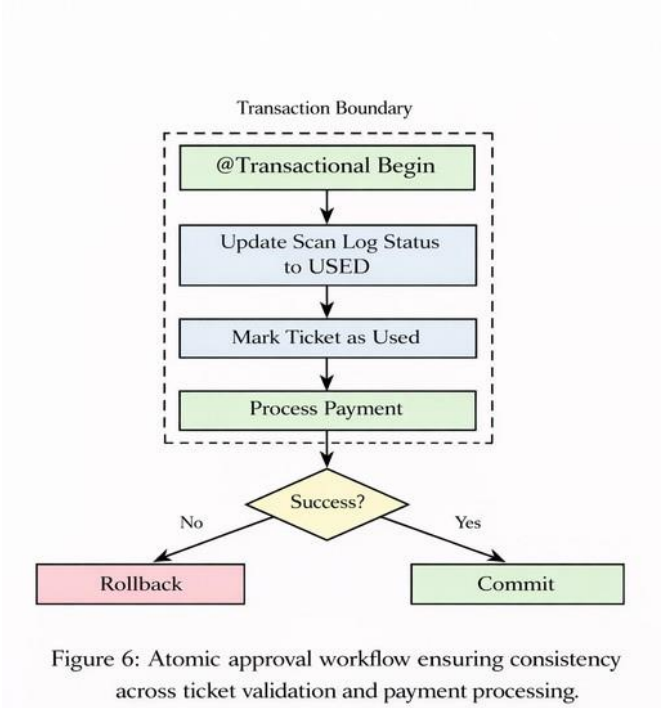

Figure 6: Atomic approval workflow ensuring consistency across ticket validation and payment processing.

4.2.3 Deferred Ticket Confirmation Design

In the proposed system, ticket confirmation is deferred until successful booking validation and administrative approval, where applicable. Ticket availability is updated only after confirmation, preventing incorrect seat allocation and ensuring consistency during failed or cancelled booking attempts.

4.2.4 Transaction-Coordinated Booking Confirmation Workflow

The system implements a transaction-coordinated workflow to manage ticket booking confirmation and QR code generation. Booking approval, ticket status update, and QR code assignment are executed within a single transactional boundary to ensure atomicity and data consistency.

If any step in the workflow fails, the transaction is rolled back automatically, preventing partial updates such as confirmed tickets without valid QR codes. This approach guarantees reliable booking confirmation, prevents data inconsistency, and ensures secure ticket validation during event entry.

4.2.5 Booking Cancellation and Retrieval

The system supports secure ticket booking cancellation to handle user-initiated or administrator-controlled rejections. When a booking is cancelled, the ticket status is updated accordingly, and the associated QR code is marked invalid to prevent unauthorized usage during event entry.

Users can retrieve and view their booking history, including confirmed, cancelled, and expired tickets, through the dashboard. Administrators can also access booking records for monitoring and reporting purposes. This retrieval mechanism ensures transparency, accurate record keeping, and efficient event management.

4.3 User and Event Management

The Event Ticket Booking System includes structured management of users and events to support efficient system operation. User management handles registration, authentication, and role assignment for customers and administrators. Event management allows administrators to create, update, and organize events with details such as date, time, location, and ticket capacity. This module

ensures accurate event information, controlled access, and smooth coordination between users, bookings, and ticket validation processes across the platform.

4.4 Entry Validation, Event Schedule, and Reporting Foundations

4.4.1 Attendance Module

The attendance module records participant entry by validating QR codes at event access points. Each successful scan updates the ticket status and logs attendance details in the database. This ensures accurate tracking of event participation, prevents duplicate entries, and provides reliable data for organizers to monitor crowd flow and attendance statistics.

4.4.2 Event Schedule Calendar

The event schedule calendar maintains information about event dates, timings, and availability. It helps organizers plan events effectively and ensures users can view accurate schedules, preventing booking conflicts and supporting smooth event management.

4.4.3 Revenue and Reporting Foundation

The revenue and reporting foundation stores ticket pricing, booking transactions, and attendance data to support financial tracking. It enables organizers to analyze ticket sales, revenue trends, and event performance, forming a base for future payment gateway and accounting system integration.

4.5 Data Transfer Objects and Exception Handling

The system uses Data Transfer Objects (DTOs) to safely exchange data between the frontend and backend, reducing exposure of internal entities. DTOs improve performance by transferring only required information. Centralized exception handling is implemented to manage validation errors, booking conflicts, and QR code verification failures. Meaningful error responses are returned through REST APIs, improving reliability, debugging, and overall user

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

experience.
Data Transfer Objects are designed separately for user registration, event details, ticket booking, and QR code validation. This separation ensures that only necessary data is transferred, reducing payload size and improving system performance.

DTOs also help in validating incoming request data before it reaches the business logic layer. Invalid or incomplete inputs are detected early, preventing unnecessary database operations and improving system robustness.

Exception handling is centralized using a global exception handler to manage runtime errors consistently across the application.

Common exceptions include invalid booking requests, unauthorized access, and QR code validation failures.

Custom exception classes are defined to represent specific error scenarios such as booking conflicts, expired tickets, and duplicate QR code scans. This improves clarity and simplifies debugging for developers.

Standardized error responses are returned in JSON format with appropriate HTTP status codes. This allows the frontend to handle errors gracefully and provide meaningful feedback to users.

4.6 Implementation Scope and Simplifications

The implementation of the Event Ticket Booking System focuses on core functionalities such as user authentication, event management, ticket booking, and QR code–based entry validation. Advanced features like online payment processing, seat selection, and third-party service integration are intentionally simplified or excluded. This controlled scope ensures clarity of design, easier testing, and effective demonstration of system architecture, validation logic, and transaction handling.

5. RESULTS AND PERFORMANCE EVALUATION

The system demonstrates reliable performance with fast booking processing, accurate QR code validation, consistent data handling, and stable operation under multiple user and ticket validation scenarios.

5.1 System Performance Characteristics

The Event Ticket Booking System demonstrates stable and efficient performance during functional and load-based testing. User operations such as event browsing, ticket booking, and QR code retrieval are processed with minimal latency. The QR code validation mechanism performs real-time verification, ensuring fast and accurate entry validation. Optimized RESTful APIs and efficient database queries reduce processing time and support concurrent user requests. Transaction management maintains data consistency during simultaneous bookings and validations. The system also handles moderate traffic without performance degradation, making it suitable for small to medium-scale events. Overall, the performance results indicate that the system is reliable, responsive, and capable of supporting real-world event management requirements.

The Event Ticket Booking System exhibits efficient performance with quick response times for ticket booking and QR code validation. RESTful APIs enable real-time processing, while optimized database queries ensure smooth operation even during multiple concurrent booking and entry validation requests.

5.2 Functional Validation Results

Functional validation confirms that all core features of the Event Ticket Booking System operate as intended. User registration, authentication, event browsing, and ticket booking functions perform accurately. QR code generation and validation correctly identify valid, used, and invalid tickets. Role-based access control ensures authorized operations for users and administrators.
Booking conflict checks and status updates function reliably across different scenarios, validating system correctness and security.

Functional validation was conducted to verify the correctness of all major system functionalities. User registration and authentication processes were tested to ensure secure access and proper role assignment. Both user and administrator workflows performed as expected.

Event management features were validated by creating, updating, and viewing events. Ticket availability was accurately reflected during booking operations, and invalid booking attempts were correctly rejected. This ensured reliable event and ticket handling.

The QR code generation and validation process was thoroughly tested at event entry points. Valid tickets were accepted and marked as used, while duplicate or invalid QR codes were rejected, preventing unauthorized access.

Booking retrieval, cancellation, and status updates were also verified across different scenarios. These tests confirmed that the system maintains data consistency, enforces business rules, and delivers reliable functionality throughout the ticket booking lifecycle.

5.3 Comparative Analysis with Manual Systems

Compared to traditional manual ticketing methods, the Event Ticket Booking System significantly reduces booking time and operational effort. Online booking eliminates long queues and physical ticket handling, while QR code validation provides faster and more secure entry. Manual systems are prone to errors, ticket duplication, and poor record keeping, whereas the proposed system ensures accurate data storage and real-time validation.
Additionally, centralized management enables organizers to monitor ticket sales and attendance efficiently, improving overall event coordination and user experience.

6. DISCUSSION

6.1 Design Validation and Key Observations

Design validation confirms that the layered architecture and

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

RESTful design effectively support secure and scalable ticket booking operations. The separation of concerns improves maintainability, while transaction-managed workflows ensure data consistency during bookings and QR code validation. Role- based access control successfully restricts sensitive operations, and the validation framework prevents duplicate bookings and unauthorized entry. Overall, the design decisions contribute to system reliability, performance, and ease of future enhancements.

The use of modern web technologies such as React and Spring Boot resulted in a responsive and reliable application. Efficient API communication ensured smooth interaction between the frontend and backend, even during concurrent booking and validation requests.

Another key observation is the effectiveness of QR code–based validation in improving entry security and reducing manual verification effort. The automated validation process minimized entry delays and enhanced the overall event management experience for organizers and attendees.

6.2 Limitations and Constraints

The Event Ticket Booking System is primarily developed as an academic prototype and focuses on demonstrating core functionalities such as event management, ticket booking, and QR code–based validation. Advanced features including online payment gateway integration, dynamic seat allocation, and automated notification services are not implemented in the current version. The system is designed to support small to medium-scale events and may require additional performance tuning and infrastructure enhancements to handle very large audiences. Real-time QR code scanning performance depends on external devices and network stability, which can affect entry validation speed. Additionally, the system currently lacks integration with third-party analytics and accounting platforms. These limitations provide scope for future enhancements and system optimization.

6.3 Alignment with Literature Findings

The proposed system aligns with existing research by automating ticket booking and using QR code validation to enhance security. Its layered architecture and RESTful design address scalability, maintainability, and validation gaps identified in earlier digital ticketing solutions.

7. CONCLUSION AND FUTURE WORK

7.1 Conclusion

The Event Ticket Booking System successfully demonstrates an effective digital solution for online event management and ticket validation. By combining web-based booking with QR code–based entry verification, the system improves security, accuracy, and user convenience. Its layered architecture and role-based access control ensure reliable operation and easy maintenance.

7.2 Future Work

Future enhancements of the Event Ticket Booking System include integrating secure online payment gateways, implementing seat selection and dynamic pricing, and adding notification services such as email and SMS alerts. The system can be extended with advanced analytics, mobile application support, and third-party integrations to handle large-scale events and improve overall event management efficiency.

ACKNOWLEDGMENT

The authors express sincere gratitude to the faculty and department for their guidance and support, and to the institution for providing resources and an encouraging environment to complete this project.

REFERENCES

[1] A. Kumar and R. Sharma, “Online Event Ticket Booking System,” International Journal of Computer Applications, vol. 175, no. 12, pp. 15–20, 2020.
[2] S. Patil, R. Joshi, and A. Kulkarni, “Web-Based Event Management and Ticket Booking System,” International Journal of Engineering Research and Technology (IJERT), vol. 10, no. 6, pp. 421–426, 2021.
[3] P. Reddy and M. Suresh, “QR Code Based Event Ticket Validation System,” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 7, no. 3, pp. 198–204, 2022.
[4] N. Choudhary, A. Khan, and S. Ansari, “Secure Online Ticket Booking Using QR Code Technology,” International Research Journal of Engineering and Technology (IRJET), vol. 9, no. 5, pp. 1120–1124, 2022.
[5] R. Mehta and K. Patel, “Design and Implementation of an Online Ticket Booking System Using Spring Boot,” International Journal of Advanced Research in Computer Science, vol. 13, no. 2, pp. 45–50, 2023.
[6] S. Verma and A. Singh, “Role-Based Web Application for Event Management,” International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 12, no. 1, pp. 233–238, 2023.
[7] T. Rao and P. Kumar, “RESTful Web Services for Event Ticket Booking Applications,” Journal of Emerging Technologies and Innovative Research (JETIR), vol. 11, no. 4, pp. 320–325, 2024.
[8] M. Gupta, R. Shah, and N. Jain, “Digital Ticketing and Crowd Management Using QR Codes,” Proceedings of the International Conference on Smart Computing and Communication, 2021, pp. 88–94.
[9] A. S. Reddy and K. Prasad, “Web-Based Event Scheduling and Ticketing System,” International Journal of Information Technology and Computer Science, vol. 14, no. 6, pp. 60–66, 2022.
[10] J. Williams and R. Brown, “Scalable Online Ticketing Systems for Large Events,” IEEE International Conference on Web Services, 2020, pp. 410–417.
[11] K. Patel and S. Desai, “QR Code–Based Secure Event Entry System,” International Journal of Advanced Computer Science and Applications (IJACSA), vol. 14, no. 3, pp. 256–261, 2023.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

BIOGRAPHIES OF AUTHORS

Seru Ravi is currently pursuing his Bachelor of Technology (B.Tech) in Computer Science and Engineering at Bonam Venkata Chalamayya Engineering College (BVCEC), Odalarevu. Interests include web technologies, database management systems, and software development. Actively participated in a team project on a web-based event management system, focusing on designing a secure online ticket booking platform with QR code–based ticket validation.

Nittala Harshit Sai Sriram is currently pursuing his Bachelor of Technology (B.Tech) in Computer Science and Engineering at Bonam Venkata Chalamayya Engineering College (BVCEC), Odalarevu. I have a strong interest in web technologies, DBMS, and software development. I worked in a team of five students to develop a web-based event management system with secure online ticket booking and QR code verification.

Sundaraneedi Surya Sai Praveen is currently pursuing his Bachelor of Technology (B.Tech) in Computer Science and Engineering at Bonam Venkata Chalamayya Engineering College (BVCEC), Odalarevu. His academic interests include web technologies, database management systems, and software development. He has actively participated in projects related to web-based event management systems, focusing on designing secure and efficient online ticket booking applications with QR code–based validation.

Pappula Veeranna is currently pursuing his Bachelor of Technology (B.Tech) in Computer Science and Engineering at Bonam Venkata Chalamayya Engineering College (BVCEC), Odalarevu. My interests include web development, database systems, and software engineering. As part of a five- member team, I contributed to the design and implementation of a web-based event management system featuring secure online ticket booking and QR code–based authentication.

Ch siva Koppeswararao is currently pursuing his Bachelor of Technology (B.Tech) in Computer Science and Engineering at Bonam Venkata Chalamayya Engineering College (BVCEC), Odalarevu. I am interested in web technologies, database management, and software development. I participated in a group project involving a web-based event management system, where we developed an online ticket booking feature with QR code validation for security.

T.V.L. Srinivasa Rao is an Assis tant Professor,M.Tech in the Department of CSE at Bonam Venkata Chalamayya Engineering College(BVCEC),Odalarevu.His research interests include software engineering, data management,and intelligent systems. He guided the development of the Hospital Appointment Booking System, ensur ing both technical and academic rigor.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)