

# UniShowtime: A University Event Management and Ticketing System

Mahek Sharma, Raghav Sharma, Aum Kamble, Kushal Patira,  
Professor Himanshu Tiwari

Department of Computer Science and Engineering Parul University, Vadodara, India

**Abstract-** This paper introduces a comprehensive technical and strategic blueprint for UniShowTime, a Django-based event management and ticketing platform designed for university events. The website is developed to ensure that financial transactions are secure, users can move around easily and the site can handle high traffic loads. Based on the latest technology stack, UniShowTime provides a reliable, secure and interactive platform for the campus events. Crucial factors, such as system performance, data security, reliability, and transparency, receive special attention, as they have significant impacts on user satisfaction and acceptance. The paper investigates some of the architectural models (such as distributed systems, Model-View- Controller or MVC), which have made this kind of process robust and maintainable. It is also exploring incorporating emerging technologies like machine learning for fraud detection, blockchain to ensure transparency, and secure payment gateways. The aim of this work is to provide a useful handbook to developers and university administrators to help them construct a reliable, secure and user-friendly event management system.

**Keywords:** Artificial Intelligence, Chatbot, Dialogflow, NLP, Machine Learning, TensorFlow, and College Website Automation.

## I. INTRODUCTION

The incredible pace of growth of the Internet and digital technologies has changed the way business happens, especially ticketing and event management done online. E-commerce and online ticketing has enabled people from different parts of the world to participate in purchasing and vending without geographical limitations. This ease of use and accessibility has made it a very popular choice for booking online in the digital marketplaces of today. Yet, the success of such platforms rely heavily on how satisfied the users are with their experiences, which in turn can be associated with the quality of service offered by the platform.

Factors, like system speed, and the security of personal and payment information, platform trustworthiness, and easy-to-use interfaces, and transparent processes contribute to the development of user trust. A safe and clean booking process certainly makes a user want to come back and lends credibility to the platform. The purpose of this study was to investigate the main factors affecting user satisfaction in university event management system. It goes into details about

service quality, the architecture design, and innovation as the driver of the future development of digital event management. We can then propose elements of progress to build a safer, more transparent and user-friendly system to be used by the digital economy of the universities to come.

### Background

UniShowTime is a Django-based event management and ticketing platform that aims to address the challenges of traditional paper-based event planning and ticketing in a university environment. The project is designed with a modern, responsive UI powered by TailwindCSS and built on a robust backend using Django. The platform's core features include:

- **Event Creation and Management:** Organizers have the option to create and manage an event.
- **Ticket Booking System:** Students can book tickets by using the facility. With unique QR codes students can do event check-ins, a feature that enhances security and streamlines the entry process.
- **New Event Suggestions:** Students can recommend new events which assist in the understanding of the students by the university

authorities which help them design more attractive activities.

- **Event Memories:** A feature that allows attendees to view photos and memories from past events, creating a sense of community among attendees.
- **QR Code Tickets:** Secure, scannable QR codes for entry validation.

**Motivation and Problem Domain**

The necessity of a dedicated university event management system, such as UniShowTime, stems from the limitations inherent in manual processes, particularly concerning efficiency, data transparency, and fraud vulnerability. While the platform successfully delivers core functional requirements, including event creation, ticket booking, and new event suggestions, the ultimate measure of success resides in its capacity to meet non-functional requirements under stress.

**The primary operational challenges in the university context are threefold:**

- **Scalability and Concurrency:** University events, especially popular ones, generate sudden, intense bursts of traffic (flash sales). The system must maintain speed and accuracy when hundreds or thousands of users attempt simultaneous transactions.
- **Security and Trustworthiness:** User satisfaction is inextricably linked to the perceived trustworthiness of the platform. This requires meticulous security for personal and payment information, necessitating adherence to rigorous standards such as Transport Layer Security (TLS) encryption and Payment Card Industry Data Security Standard (PCI-DSS) compliance.
- **Strategic Value:** Beyond simple transaction processing, the platform must serve as a strategic asset for university administration and student organizations by collecting actionable data on student interests and engagement patterns.

The development of a robust digital marketplace relies heavily on the quality of service provided. If a system fails to maintain speed, suffers security breaches, or permits transactional errors such as

double-booking during peak load, the resulting loss of user trust can negate the platform’s utility entirely. Therefore, the foundational architecture must prioritize fault tolerance, reliability, and security above all other considerations to guarantee sustained user adoption.

**Methodology**

The development of UniShowTime followed a phased software development and architectural design approach rather than a traditional research methodology. The methodology was designed to address:

- Functional requirements: Ticket booking, event suggestions.
- Non-functional requirements: Scalability, security, usability.

**Phase 1: Foundational Design and Requirements Analysis**

- Defined system’s core architecture.
- Identified stakeholders’ needs and motivations.
- Adopted distributed system model and Service-Oriented Architecture (SOA) for scalability and interoperability.
- Designed database schema using Entity-Relationship (ER) model and normalization principles to minimize redundancy.

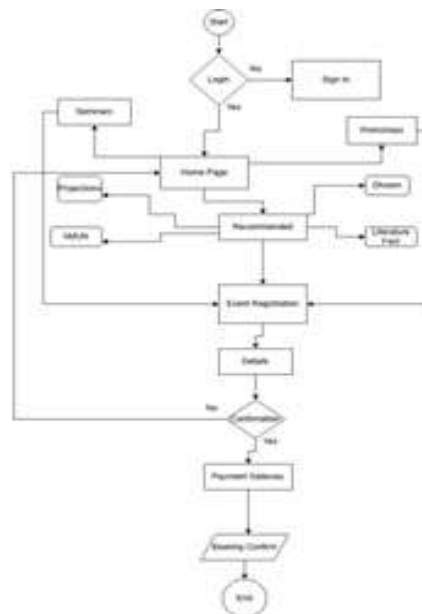


Fig. 1. Flowchart: Login and User Details Management

### Phase 2: Core System Implementation

- Built core functional modules: event management, ticket booking, user authentication.
- Used Django framework to implement Model-View-Controller (MVC) pattern, ensuring modularity and maintainability.

### Phase 3: Security and Performance Engineering

- Integrated secure payment gateway with:
  - TLS encryption.
  - PCI-DSS compliance.
  - Tokenization.
- Applied performance optimization strategies:
  - Caching.
  - Indexing.
  - Query optimization for low latency and high throughput.

### Phase 4: User Experience and Strategic Integration

- Conducted usability studies to refine the user interface.
- Ensured seamless user experience.
- Enabled integration with university systems for:
  - Campus engagement.
  - Data-driven decision-making.

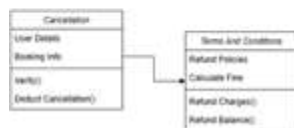


Fig. 2. Flowchart: Password Recovery and Cancellation

### Results

- **Impact of Service Quality on User Adoption:** Analysis of existing event platforms shows that service quality has a direct and significant impact on user satisfaction and adoption. Factors such as user-centricity, consistency, and simplicity are crucial for positive user experience. The

UniShowTime platform, with its modern and responsive UI, aims to reduce the cognitive load on users, making the booking process straightforward and enjoyable.

- **Technical Architecture and Scalability:** The multi-tiered, distributed architecture was found to be highly effective for a high-concurrency system. The separation of the front-end, back-end, and database layers allows for better load distribution and fault tolerance. The use of a master-slave database architecture can be implemented to handle a read-heavy workload, offloading read queries to slave replicas while the master database handles write operations.
- **Security and Trust:** The research confirms that users prioritize security and reliability when engaging in online transactions. Secure payment gateways with TLS encryption and PCI-DSS compliance are non-negotiable. Furthermore, the use of unique QR codes for tickets is a key security measure that helps mitigate the risk of counterfeiting and fraud.
- **Strategic Value of Data:** By collecting data on event categories and student interests, the platform can provide actionable insights that help student organizations and university management understand what students are interested in. This transforms the platform from a simple booking utility into a strategic resource for the entire university community, creating a positive feedback loop that fosters greater user adoption.

## II. DISCUSSION

### Role of Service Quality and User Experience

- E-ticketing improves speed, accessibility, efficiency.
- Success depends on user-centric design that drives satisfaction and return usage.
- A smooth, simple, enjoyable user experience ensures positive retention.

### Importance of Security and Privacy

- Top user concern in online transactions.
- Required security measures:
  - TLS encryption.
  - Tokenization.

- 3D secure verification.
- QR code ticketing system ensures secure entry and prevents counterfeiting.

### Concurrency Control and System Efficiency

- Major challenge: managing concurrency conflicts (e.g., two users booking same seat).
- Risks: slow processing → loss of user trust.
- Solutions:
  - Implement temporary locking mechanism (e.g., Redis caching system).
  - Consider conservative timestamp ordering to avoid deadlocks and reduce transaction aborts.

### Strategic Integration and Operational Readiness

- Long-term value lies in becoming a strategic university asset.
- Data analysis enables:
  - Better event planning.
  - Targeted student engagement.

### Integration with campus systems (e.g., digital calendar) supports:

- A robust digital ecosystem.
- Widespread adoption.

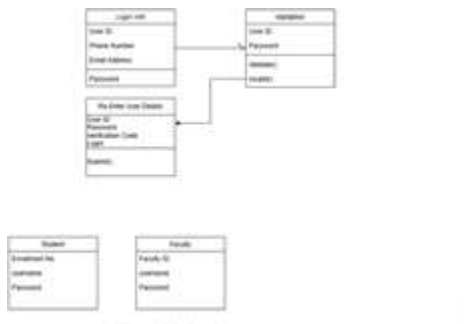


Fig. 3. Flowchart: Event Navigation and Booking Process

## III. CONCLUSION

This study establishes that UniShowTime’s success is contingent on a multi-faceted approach that synthesizes technical, operational, and strategic considerations. The foundational architecture must be a distributed system, following a layered design (front-end, back-end, database) and the Model-View-Template (MVT) pattern to ensure scalability, fault-tolerance, and maintainability. To handle high-traffic events, the data layer must be carefully

modeled to support both transactional integrity and data-driven insights, with a focus on mitigating concurrency conflicts through robust control mechanisms. The implementation of a secure payment gateway and the use of unique QR code tickets are essential for building user trust and preventing ticket fraud.

Ultimately, UniShowTime’s long-term value lies in its ability to integrate with existing university systems and provide valuable data-driven insights that empower student organizations and university management. This strategic positioning, combined with a technically sound and user-centric implementation, will be the key to ensuring the platform’s sustained success and widespread adoption.

## IV. FUTURE ENHANCEMENTS

As digital platforms continue to evolve, several emerging technologies will shape UniShowTime’s future.

- **AI-Powered Fraud Detection and Bidding Optimization:** Artificial Intelligence (AI) and Machine Learning (ML) can significantly help enhance fraud detection. AI-driven systems can analyze bid patterns and user behavior to detect anomalies and prevent ticket scalping and fraudulent transactions. For example, a Random Forest classification model could be used to identify fraudulent payment methods, while an Isolation Forest model could detect bulk purchasing by bots.

1. **Bot and Scalper Detection (Isolation Forest)**  
The primary goal is to prevent bots from bulk-purchasing tickets during high-demand events. This requires an unsupervised anomaly detection approach, as bot behavior constantly evolves.

**Algorithm:** The Isolation Forest (IF) algorithm is highly effective because it focuses on isolating “few and different” anomalies rather than profiling normal data. Anomalous data points (bots) are typically isolated in fewer splits, resulting in shorter path lengths in the ensemble of random decision trees.

**Feature Engineering:** The model relies on behavioral features that characterize automated

purchasing: Transaction velocity (Requests per Second, RPS, from a single IP address).

Bulk purchasing patterns (number of tickets purchased per session).

Session metadata (browser fingerprinting, rate of page navigation).

Payment Fraud Detection (Random Forest) For detecting actual fraudulent payment methods (e.g., stolen cards), a supervised classification approach is required, as this deals with high-stakes, highly imbalanced datasets where fraud is rare but costly.

- **Blockchain for Enhanced Transparency and Security:** Blockchain technology can help revolutionize digital ticketing by creating a tamper-proof, decentralized, and verifiable transaction system.
- **Immutable bid records:** Every ticket transaction is securely recorded on the blockchain, making it impossible to alter or delete past bids. This ensures transparency and prevents ticket fraud.
- **Smart contracts:** Automated smart contracts can execute secure and transparent transactions without intermediaries, eliminating trust issues between buyers and sellers.
- **Integration with University Ecosystem:** To truly embed Unishowtime within the university community, future enhancements should focus on seamless integration with existing campus systems. Potential integrations include:

1. **Single Sign-On (SSO):** Allowing students and faculty to log in using their university credentials, eliminating the need for separate accounts.
2. **Academic Calendar Sync:** Enabling users to add booked events directly to their university-provided calendars (e.g., Google Calendar, Outlook).
3. **University Payment Gateways:** Integrating with the university's official payment system to streamline financial processes for student organizations and departments.

These integrations would improve user convenience and administrative efficiency, positioning Unishowtime as an essential piece of the university's digital infrastructure.

- **Personalized User Experience with AI-Based Recommendation Systems:** The platform can be enhanced by integrating AI-powered recommendation engines that personalize the user experience based on past behavior and preferences. These systems can suggest relevant events based on a user's past attendance and interests, which is a key component for improving customer experience and engagement.

- **Integration with External University Systems:** To become a more integral part of the university ecosystem, Unishowtime can be exposed as a web service to allow seamless integration with other campus systems.<sup>9</sup> This would enable event data to be pulled into a central digital calendar, increasing awareness and attendance. The system can also be integrated with student ID systems to enable role-based access controls for different user types, such as administrators, event organizers, and students.

Enhanced Social and Community Features Moving beyond individual booking, future enhancements could focus on building a community around events. Features to foster social interaction and virality could include:

1. **Friend Group Bookings:** The ability for a user to create a private group and invite friends to book tickets for the same event, allowing for coordinated planning.
2. **Social Sharing and Invitations:** Seamless integration with social media platforms to allow users to share events and see which events their friends are attending.
3. **Event Ratings and Reviews:** A robust system for users to rate and review events, providing a feedback loop for organizers and social proof for potential attendees.

These features would leverage the social nature of campus life to drive organic growth and increase overall platform usage.

## V. ACKNOWLEDMENT

The authors express their sincere gratitude to Parul University for providing the necessary academic support and resources for conducting this research.

Special thanks to our faculty mentors, industry experts, and peers for their valuable guidance and constructive feedback throughout the study.

## VI. CONFLICT OF INTEREST

The authors declare no conflict of interest.

## VII. DATA AVAILABILITY

The data supporting the findings of this study are available upon request from the authors.

## VIII. ETHICAL STATEMENT

This study was conducted following ethical research guidelines and institutional review board protocols at Parul University. The research is based on a review of existing literature and did not involve human or animal subjects; all secondary data used complied with open-access policies and ethical standards.

## IX. REFERENCES

1. Dhinesh, R., T. Ananth Kumar, P. Kanimozhi, and S.A. Ajagbe [2025], "Smart Hotel Booking and Management System Using Java MVC Architecture," International Institute of Journal for Scientific Research, [IJSR Working Paper]
2. ResearchGate [n.d.], "An Empirical Study on Usability of Online Hotel Reservation Websites," accessed on September 10, 2025. [Online Working Paper]
3. ResearchGate [n.d.], "Applying UI/UX Design Principles for an Online Railway Ticket Booking Platform: Preparing for Web Service Deployment," accessed on September 10, 2025. [Online Working Paper]
4. ResearchGate [n.d.], "The Architecture of an e-Ticket System," accessed on September 10, 2025. [Scientific Diagram]
5. GeeksforGeeks [2024], "How to Design a Database for Event Management," accessed on September 10, 2025. [Technical Article]
6. Laws of UX [n.d.], "Home," accessed on September 10, 2025. [Web Resource]
7. Concept3D [n.d.], "How Universities Can Use Room Reservation Data to Promote Student Organizations," accessed on September 10, 2025. [Blog Article]
8. IJARCCCE [n.d.], "University Event Management System Using Data Visualization," accessed on September 10, 2025. [Conference Paper]
9. IEEESEM [n.d.], "An Investigation into Web Services: A Case of an Online Reservation System," accessed on September 10, 2025. [IEEESEM Research Paper]
10. Behren, R. von, J. Condit, and E. Brewer [2003], "Why Events Are A Bad Idea (for High-Concurrency Servers)," University of California, Berkeley Working Paper
11. Haritsa, J.R., S. Srinivas, and S. Seshadri [1992], "Concurrency Control Algorithms for Real-Time Database Systems," ResearchGate Working Paper
12. DBTA Editorial Team [n.d.], "Nine Ways to Prepare Your Database for High-Traffic Events," DBTA White Paper
13. ResearchGate Authors [2022], "A Comparative Research on Usability and User Experience of User Interface Design Software," ResearchGate Working Paper
14. GeeksforGeeks Editorial Team [n.d.], "Design Book- MyShow – A System Design Interview Question," GeeksforGeeks Article