

TicketEase: A Dynamic Movie Ticket Booking System

Authors:

Harshvardhan Barai (Student, Department of Computer Science & Engineering, Parul Institute of Engineering & Technology, Vadodara, Gujarat, India)

Asst. Prof. Pathan Bilalkhan R. (Assistant Professor, Department of Computer Science & Engineering, Parul Institute of Engineering & Technology, Vadodara, Gujarat, India)

Abhishek Raimangiya (Student, Department of Information & Technology, Parul Institute of Engineering & Technology, Vadodara, Gujarat, India)

Asst. Prof. Sonali Kori (Assistant Professor, Department of Information & Technology, Parul Institute of Engineering & Technology, Vadodara, Gujarat, India)

ABSTRACT

TicketEase is an online ticket booking application designed for seamless and efficient movie ticket reservations. The platform provides a user-friendly interface with a no-code approach, enabling users to book movie tickets, select seats, and make payments without technical expertise. Built using Next.js for the frontend, Svelte for interactive widgets, and Node.js with MySQL for backend processing, TicketEase ensures high performance and smooth data handling. The system integrates Stripe and PayPal for hassle-free transactions and leverages Google reCAPTCHA for security. Additionally, webhook integration allows real-time updates and automation with external applications. TicketEase is a scalable, cost-effective, and user-centric solution for modern ticket booking needs.

Keywords: no-code booking, payment integration, workflow automation, webhook support, user-friendly interface, movie ticket reservation.

INTRODUCTION

The digital era has revolutionized the way users book movie tickets, with traditional methods being replaced by online platforms that offer ease, security, and flexibility. However, existing ticket booking applications often suffer from high subscription costs,

sluggish performance, and limited customization. TicketEase addresses these challenges by providing a fast, scalable, and developer-friendly solution that enhances the movie ticket booking experience.

Built with Next.js, Svelte, and Node.js, TicketEase offers an optimized frontend and backend architecture. The system utilizes Redis caching and a queue-based background processing mechanism to handle high-volume transactions efficiently. This paper explores the architecture, implementation, and advantages of TicketEase, showcasing its effectiveness in improving the online ticket booking experience.

METHODOLOGY

The architecture of TicketEase is designed for performance, scalability, and an enhanced user experience. It comprises multiple components that ensure smooth transaction handling and data integrity.

Frontend (Client-Side) TicketEase consists of two major frontend components:

1. **Main Application** : The primary web interface allows users to browse movies, select showtimes, book tickets, and manage reservations. It is built using Next.js, which

provides server-side rendering (SSR) for better performance and SEO optimization.

2. **Widget** : This embeddable widget allows third-party websites (cinema websites, affiliates) to integrate TicketEase's booking system seamlessly. The widget, developed in Svelte, ensures high performance with minimal runtime overhead.

Both components communicate with the backend via RESTful APIs to fetch and store data dynamically.

Backend (Server-Side Processing) The backend, built with Node.js, serves as the core processing unit for the application. It is responsible for:

- User authentication and authorization.
- Processing ticket bookings and managing seat availability.
- Handling payments securely through Stripe and PayPal.
- Implementing security measures such as Google reCAPTCHA.

Data Handling & Caching TicketEase uses Redis as a caching layer to optimize data retrieval and minimize database load. This improves response times, especially during high-traffic periods.

- When a user queries available seats, the system first checks Redis before making a database call.
- If the data is not cached, the MySQL database is queried, and the results are stored in Redis for future access.

Asynchronous Processing (Queue & Worker System)

Operations such as sending email confirmations, processing refunds, and handling webhooks are managed using a queue-based system.

- Tasks are placed into a queue for background processing.
- Backend workers continuously process these tasks without affecting the real-time booking experience.

EXISTING SYSTEM AND PROPOSED SYSTEM

Existing Systems: Current online ticket booking platforms face challenges such as:

- High subscription costs.
- Performance bottlenecks under high traffic.
- Limited customization and third-party integration.

Proposed System: TicketEase overcomes these limitations with:

- A scalable architecture built with Next.js, Svelte, and Node.js.
- Redis caching and queue-based background processing for efficiency.
- API-driven design for easy integration with cinema management systems.

USE CASES AND IMPACT

Primary Use Cases:

- **Cinema Websites:** Allowing direct ticket booking with real-time seat availability.
- **Event Organizers:** Enabling ticket sales for movie premieres and special screenings.
- **Affiliates & Partners:** Embeddable widgets for third-party sales.
- **Customer Management:** Integration with loyalty programs and personalized recommendations.

Impact:

- **Enhanced Performance:** Faster booking experiences with optimized frontend technologies.
- **Scalability:** Queue and caching mechanisms ensure smooth operations under high demand.
- **User Experience:** Mobile-friendly, intuitive UI for effortless booking.

- **Security & Data Privacy:** Encrypted transactions and Google reCAPTCHA to prevent fraud.
- **Cost-Effectiveness:** Lower operational costs compared to traditional ticketing systems.
- **Seamless Integration:** Webhook and API support for easy embedding and automation.

CONCLUSION

TicketEase presents a cutting-edge, scalable, and efficient solution for online movie ticket booking. Leveraging modern web technologies, it ensures optimal performance, security, and ease of integration. By addressing the limitations of existing ticket booking platforms, TicketEase provides a cost-effective and flexible alternative for cinemas, event organizers, and affiliates.

ACKNOWLEDGEMENTS

I express my gratitude to **Asst. Prof. Pathan Bilalkhan R. & Asst. Prof. Sonali Kori** for guidance and extend

my appreciation to **1Stop Team** for supporting the development of TicketEase.

REFERENCES

- [1] R. Kumar, S. Sharma, and P. Gupta, "Enhancing Web-Based Booking Applications using Next.js and Svelte for Scalable Systems," *International Journal of Computer Science and Software Engineering*, Vol. 25, Issue 4, 2023, pp. 102-110.
- [2] A. Patel and V. Srinivasan, "Optimizing Ticket Management with Node.js and MySQL: A Performance Analysis," *Journal of Software Engineering and Development*, Vol. 18, Issue 3, 2022, pp. 89-97.
- [3] J. Lee, H. Kim, and T. Park, "Efficient Data Storage and Retrieval Mechanisms in Web Applications using MySQL and Redis," *International Journal of Data Science and Applications*, Vol. 20, No. 2, 2021, pp. 134-145.
- [4] M. Ramesh and K. Varun, "A Comparative Study on Traditional vs. Modern Online Booking Systems: Performance, Security, and Scalability," *International Journal of Web Technologies*, Vol. 15, Issue 1, 2020, pp. 50-62.