EVENT MANAGEMENT AND TICKET BOOKING

Name : Bibhuti Bhusana Ojha Regd.No. 2101298209 Email : bbojha2021@gift.edu.in
Name : Bibhudarshan Sutar Regd. No. 2101298170 Email:bibhudarshan2021@gift.edu.in
4th Year, Computer Science & Engineering (AI) , GIFT, Bhubaneswar Affiliated to : Biju Patnaik University of Technology, Rourkela, Odisha .
***Guided by:* Dr. Sujit kumar Panda**, Professor, Department of CSE, GIFT, Bhubaneswar, BPUT , Rourkela , Odisha .

Abstract :
The Event Management & Ticket Booking System is a dynamic web application designed to streamline event browsing, ticket purchasing, and administrative event management. Developed using Flask and Firebase, the system supports user-friendly interactions such as registration, event discovery, ticket booking, and viewing booking history. On the administrative end, event organizers can create, edit, and delete events, manage ticket pricing, and monitor analytics. Key features include seat selection, e-ticket generation, refund processing, and real-time booking insights. With robust Firebase authentication and Firestore integration, the system ensures secure user data handling and seamless cloud storage for assets like event posters. This project effectively addresses the needs of both event attendees and organizers by combining usability with functionality, delivering a scalable and secure platform for modern event ticketing. Confirmation.

Keyword :
HTML, CSS, JS , PYTHON FLASK,FIREBASE

1. INTRODUCTION

The growing demand for digitized solutions in the event industry has led to the evolution of online event management platforms. The Event Management & Ticket Booking System aims to bridge the gap between event organizers and attendees by offering a comprehensive, web-based solution. Developed using Flask as the backend framework and Firebase as the backend-as-a-service (BaaS), this application provides an end-to-end solution for discovering events, booking tickets, and managing bookings. The system caters to both users and administrators through dedicated interfaces and role-based access. Enhanced security, real-time updates, and user authentication bolster the system's reliability. The integration of modern frontend and backend technologies ensures responsiveness and efficiency, while cloud-based infrastructure supports scalability. Overall, this system provides a modern and streamlined alternative to traditional event ticketing and management practices.

2. LITERATURE REVIEW

Previous studies and systems in event management often relied on standalone desktop applications or rudimentary websites with limited interactivity and scalability. Traditional models lacked real-time updates, integrated authentication, and seamless user experience. Cloud computing and web frameworks have revolutionized the way events are managed online. Frameworks like Flask allow rapid development and integration of services, while Firebase offers real-time database, authentication, and storage services in one package.

Prior literature also highlights challenges in booking systems such as overbooking, lack of user personalization, and security vulnerabilities. This system addresses those concerns through real-time seat availability monitoring, secure user roles, and a user-centric design. Moreover, studies emphasize the importance of responsive design and easy navigation, both of which are achieved through modern frontend technologies like HTML5, CSS3, and JavaScript in this implementation.

3. SYSTEM DESIGN

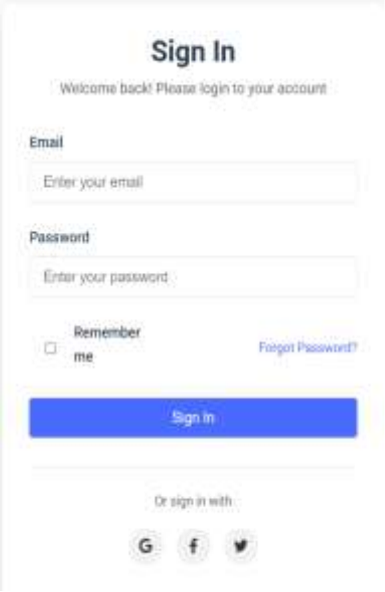
The system is architected using a modular design that separates user and admin functionalities. At its core, the Flask application serves as the backend, managing API endpoints and routing, while Firebase handles user authentication, data storage, and real-time updates.

The frontend is developed with standard web technologies and connects seamlessly with the backend. HTML templates are organized under separate directories to distinguish between admin and user views. Event data is stored in Firestore, allowing for dynamic updates and easy scalability. Security is enforced using Firebase Authentication and CSRF protection, ensuring role-based access and secure sessions. Event posters are hosted using Firebase Storage.

The booking process is transactionally managed to prevent duplicate or overlapping seat bookings. Overall, the system follows best practices in MVC (Model-View-Controller) design for maintainability and scalability.

4. IMPLEMENTATION

The application was implemented using Flask, with Firebase handling backend services like authentication, storage, and database management. The development process began with setting up the project structure, including separate directories for static files, templates, and server-side logic. Firebase was configured by generating a service account key and enabling Firestore, Authentication, and Storage. User registration and login functionalities were built with Firebase Authentication, followed by the integration of event listing and booking modules. Admin functionalities were implemented to allow creation and management of events, ticket categories, and booking statistics. The system includes CSRF tokens and form validation to enhance security. Routes are defined for user and admin dashboards, and RESTful APIs manage data communication. Testing involved simulating multiple booking scenarios and admin actions to ensure data integrity and use experience.



Sign In

Welcome back! Please login to your account

Email

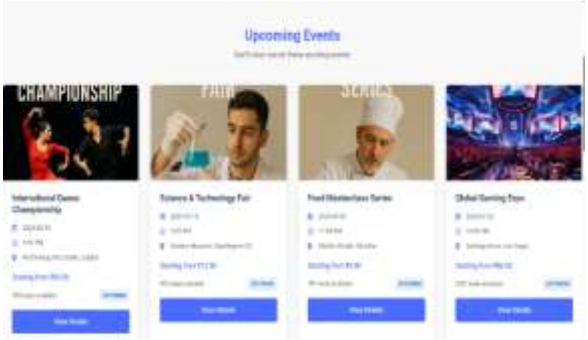
Enter your email

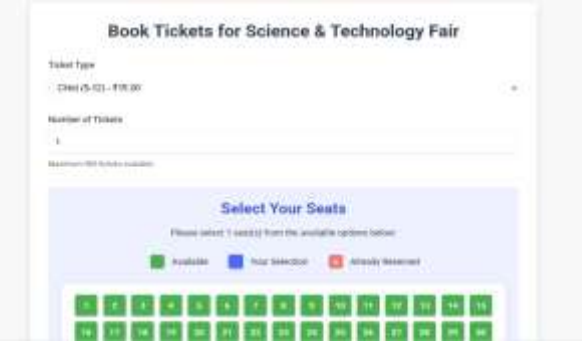
Password

Enter your password

□ Remember me

Forgot Password?





Book Tickets for Science & Technology Fair

Ticket Type

Child (5-12) - ₹[illegible]

Number of Tickets

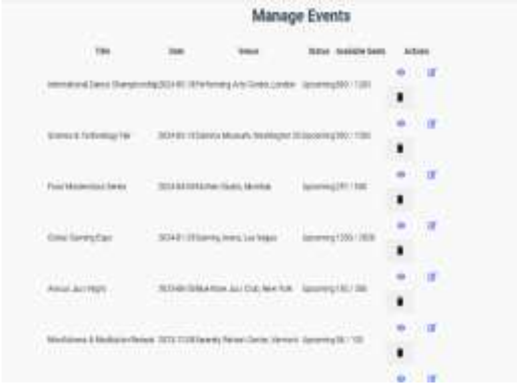
1

[illegible]

Select Your Seats

Please select 1 seat(s) from the available options below

🟩 Available 🟦 Your Selection 🟥 Already Reserved



Manage Events

Title	Date	Venue	Status	Available Seats	Actions
International Dance Championship	[illegible]	Performing Arts Centre, London	Upcoming	[illegible]	
Science & Technology Fair	[illegible]	Science Museum, Washington DC	Upcoming	[illegible]	
[illegible] Series	[illegible]	[illegible] Studio, Mumbai	Upcoming	[illegible]	
Global Gaming Expo	[illegible]	Gaming Arena, Las Vegas	Upcoming	[illegible]	
Annual Jazz Night	[illegible]	Blue Note Jazz Club, New York	Upcoming	[illegible]	
Mindfulness & Meditation Retreat	[illegible]	Serenity Retreat Center, Vermont	Upcoming	[illegible]	

5. RESULTS

The final system demonstrated robust performance across both user and admin modules. Users were able to register, browse events, select seats, and receive confirmation via the platform. Real-time updates from Firebase ensured that seat availability reflected actual bookings, eliminating double bookings. Admin users successfully managed events, including creating and editing event information and monitoring analytics such as ticket sales and booking trends.

The interface was intuitive and mobile-friendly, enhancing accessibility. Security features like input sanitization, CSRF protection, and secure sessions functioned effectively during testing. Performance benchmarks showed fast response times and reliable database interactions. The modular codebase also made it easy to maintain and scale the application. Overall, the system met its objectives of providing a functional, secure, and scalable event management and ticketing platform.

6. CONCLUSION

The Event Management & Ticket Booking System successfully addresses the modern needs of digital event coordination and ticketing. By leveraging Flask for the application logic and Firebase for backend services, the system delivers a scalable, secure, and user-friendly solution for event discovery and management. The design ensures smooth interaction for both attendees and organizers, with features like seat selection, e-ticket generation, and booking history adding value to the user experience. Administrators benefit from a centralized control panel to manage events and monitor performance analytics. Security implementations like authentication, role-based access, and CSRF protection ensure robust protection. The system’s architecture allows future enhancements such as payment gateway integration or AI-based event recommendations. This project showcases the effective integration of web development technologies in building a comprehensive event solution.

7. ACKNOWLEDGMENT

We would like to express our sincere gratitude to everyone who contributed to the successful completion of the Event Management & Ticket Booking System project. Our heartfelt thanks go to our mentors and instructors for their constant support, invaluable feedback, and guidance throughout the development cycle. We also extend our appreciation to the open-source community for providing robust tools and frameworks like Flask and Firebase, which formed the foundation of this project. Special thanks to our peers and testers who offered insights during the testing phase, helping us refine the system for optimal performance and usability. We acknowledge the documentation and libraries that made integration and implementation seamless. Lastly, we are thankful for the collaborative spirit and dedication of our development team, whose hard work and innovative thinking brought this project to fruition. Without this collective effort, building a secure, efficient, and user-friendly system would not have been possible.

8. REFERENCES

https://flask.palletsprojects.com/en/stable/
https://www.freecodecamp.org/news/how-to-build-an-event-booking-app-using-html-css-javascript-and-firebase/
https://github.com/topics/event-management-system
https://www.geeksforgeeks.org/design-movie-ticket-booking-system-like-bookmyshow/
https://github.com/topics/event-management?l=css&o=asc&s=stars