



College Event Registration System with Analytics , Payment Gateway Integration and QR-Code based Entry

¹P.V.Sai Charan, ²Syed Hyder Ali , ³Raavi.Harshith sai Kumar , ⁴Dr G.Reshma

^{1,2,3}B.Tech Student, ⁴Sr. Assistant Professor

^{1,2,3,4}Information Technology,

^{1,2,3,4}Prasad V. Potluri Siddhartha Institute of Technology, Vijayawada, India

ABSTRACT: In college fests and technical events, the registration-to-entry workflow often breaks down due to manual steps: students register but payment confirmation takes time, coordinators maintain spreadsheets, and entry validation is slow or unreliable. These gaps create long queues at gates, allow duplicate/unauthorized entry, and make it hard for organizers to track revenue and attendance in real time. This project proposes a College Event Registration System with Analytics , Payment Gateway Integration and QR-Code based Entry (PVPSIT Event Manager) that digitizes the entire lifecycle—event creation, student registration, secure online payment, QR pass generation, and QR-based check-in—through a unified web platform. The system uses a role-based architecture where students can browse upcoming events, register (including optional sub-events), pay online via Razorpay Orders + signature verification, and receive a unique QR entry pass generated only after successful payment verification. Admins can manage events, scan QR codes at the venue using a web-based scanner, and mark attendance instantly while preventing reuse through single-use validation and attendance locking. To support event decision-making, the platform provides an analytics dashboard showing registration counts, paid vs unpaid conversions, attendance rate, and revenue summaries, including breakdowns by department/branch and semester, with downloadable CSV reports. By replacing the previous manual payment confirmation process with verified gateway payments and enabling QR-based entry control, the system improves payment authenticity, reduces entry-time congestion, strengthens auditability, and provides actionable insights for organizers.

Keywords: College Event Management, Event Registration, Razorpay Integration, Payment Verification, QR-Code Entry Pass, Attendance Tracking, Analytics Dashboard, CSV Reporting, React, Spring Boot, MySQL, JWT, ZXing QR, html5-qrcode, Chart.js.

I. INTRODUCTION

College fests and technical events usually have a large number of participants, multiple events happening at the same time, and strict deadlines for registration, payment, and entry. Many colleges still use partly manual methods for the entire process. These methods include Google Forms, spreadsheets, WhatsApp screenshots, and handling cash offline. While they might work for smaller events, they become inefficient and prone to mistakes as the number of participants and events grows. Common problems include:

- Delays in payment verification because of manual checks (like reviewing UPI screenshots or bank statements).
- Duplicate or fake payment claims along with mismatched registration records.
- Slow entry management at the venue, since names and roll numbers are checked manually.
- No real-time updates for organizers about registrations, payments, attendance, and revenue.
- Challenges in producing reports based on department or semester for analysis and documentation.

To fix these issues, we created a College Event Registration System with Analytics, Payment Gateway Integration, and QR Code-based Entry. This system is a complete web platform that digitizes the workflow: it handles event publishing, student registrations, secure online payments, automatic confirmations, QR code generation for entry passes, QR scanning at the gate for attendance, and an analytics dashboard for reporting. A major improvement in this project is the move from manual payment confirmation, where organizers verify payments by hand, to a Razorpay-integrated payment process that uses order creation and signature verification for secure payment confirmation. After payment is verified, the system automatically generates a unique QR code for registration, which serves as the student's entry pass. At the venue, admins can scan the QR code with a scanner interface, marking attendance and preventing the same QR from being used multiple times.

The main goals of this system are to: - Shorten the time from registration to confirmation by automating payment verification. - Provide quick entry using QR codes, reducing crowd congestion at gates. - Prevent duplicate entries and enhance event security through server-side validation. - Give organizers real-time insights into registrations, payments, attendance, and revenue. - Create downloadable reports (CSV) for departmental and event documentation and audits.

II. LITERATURE SURVEY

The rapid growth of digital technologies has significantly influenced event management systems, particularly in automating registration, payment processing, and attendance tracking. Several researchers and platforms have contributed to the development of efficient event management solutions.

Traditional event registration systems primarily relied on manual processes such as Google Forms, spreadsheets, and offline payment verification. These systems were simple but lacked integration, leading to issues like delayed payment confirmation, data inconsistency, and difficulty in tracking participants. Manual verification methods, such as checking UPI screenshots, often result in errors and fraudulent claims, thereby reducing system reliability.

With the advancement of web technologies, modern systems have adopted **online event management platforms** that integrate registration and payment modules. Payment gateway systems such as Razorpay provide secure transaction mechanisms using order creation and signature verification, ensuring authenticity and reducing disputes. According to Razorpay documentation, secure payment verification improves transaction reliability and minimizes fraud.

Another significant development in event management is the use of **QR (Quick Response) codes** for entry validation. QR-based systems provide a fast and contactless method for verifying participants. Research on QR technology shows that it enhances security by enabling unique identification and reduces entry time compared to manual verification methods. Libraries such as ZXing enable efficient QR code generation and decoding, while html5-qrcode supports browser-based real-time scanning.

In addition, **analytics and reporting systems** have become an essential component of modern platforms. Tools like Chart.js enable visualization of event data such as registrations, payments, attendance, and revenue. Studies indicate that real-time analytics helps organizers make informed decisions and improve event planning. Database systems like MySQL further support structured data storage and efficient query processing for reporting purposes.

Frameworks such as Spring Boot and React have also played a crucial role in building scalable web applications. Spring Boot simplifies backend development through REST APIs and security features, while React enables dynamic and responsive user interfaces [4]. Additionally, Spring Security with JWT authentication ensures secure access control and role-based authorization in such systems.

Despite these advancements, many existing systems still lack a **fully integrated solution** that combines registration, secure payment verification, QR-based entry, and analytics into a single platform. Most solutions focus on individual components rather than providing an end-to-end system.

III. PROPOSED SYSTEM

The proposed system is a **College Event Registration System with Analytics, Payment Gateway Integration, and QR-Code Based Entry** that automates the complete event management process through a centralized web platform. It enables administrators to create and manage events, while students can register and make secure online payments using Razorpay with verified transaction processing. After successful payment verification, the system generates a unique QR code that serves as an entry pass. During the event, the QR code is scanned to mark attendance and prevent duplicate entries through single-use validation.

Additionally, the system provides a real-time analytics dashboard that displays registration data, payment status, attendance, and revenue insights, along with downloadable reports. This integrated approach improves efficiency, enhances security, reduces manual errors, and ensures a smooth and reliable event management experience.

IV. METHODOLOGY

The methodology outlines the steps involved in developing a system for predicting multiple diseases using machine learning. The approach consists of several stages, including data collection, data preprocessing, model development, and evaluation of the model's performance.

4.1 Dataset Description

The system collects data from multiple sources including student registrations, event details, payment transactions, and attendance records. User data such as name, roll number, department, and event selections are collected during registration. Payment data is obtained through Razorpay integration, including transaction ID, payment status, and order details. Attendance data is captured through QR code scanning at the event venue. All data is stored in a structured format within a MySQL database for further processing and analysis.

4.2 Data Preprocessing

The collected data is cleaned and validated before processing. This includes removing duplicate entries, verifying payment status, and ensuring consistency between registration and payment records. Invalid or incomplete data is filtered out. Data is then structured into proper database tables, linking users, events, payments, and attendance. Preprocessing ensures accuracy and reliability for analytics and system operations.

4.3 Model Description

The system follows a **full-stack web application model** based on a three-tier architecture. The frontend is developed using React for user interaction, while the backend uses Spring Boot to handle business logic, authentication, and APIs. MySQL is used as the database for storing all system data. The model integrates external services such as Razorpay for secure payment processing and QR code libraries (ZXing and html5-qrcode) for generating and scanning QR codes.

4.4 Prediction Approach

Although the system is not based on machine learning, it uses analytical logic to generate insights such as registration trends, payment conversion rates (paid vs unpaid), attendance percentages, and revenue analysis. These insights help organizers predict event participation, estimate revenue, and make informed decisions for future events.

4.5 System Flow

The system flow begins with admin event creation, followed by student registration. After registration, the student initiates payment through Razorpay. Once the payment is successfully verified, a QR code is generated for the participant. During the event, the admin scans the QR code to validate entry and mark attendance. All actions are updated in real-time and reflected in the analytics dashboard.

4.6 System Architecture

The system is designed using a **three-layer architecture** consisting of the presentation layer (React frontend), application layer (Spring Boot backend), and data layer (MySQL database). The frontend handles user interaction, the backend processes requests and manages logic, and the database stores all records. External integrations include Razorpay for payments and QR libraries for entry validation, ensuring a secure and scalable system.

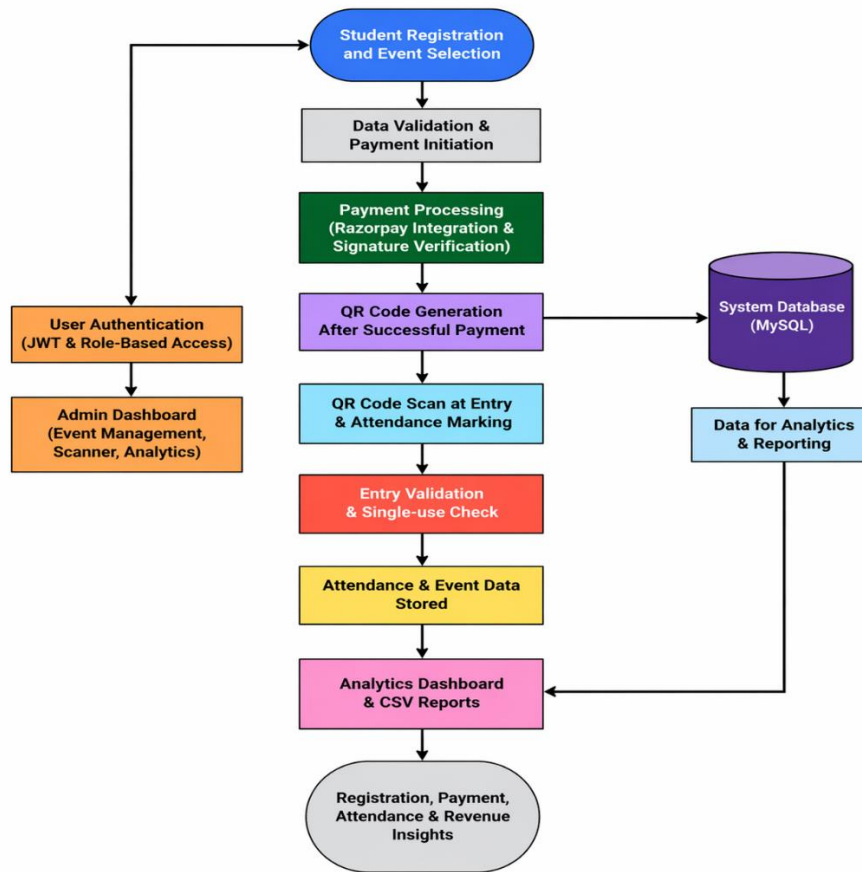


Fig. 1: System Architecture

V. RESULTS

The proposed system demonstrates significant improvements in college event management by automating registration, payment verification, and entry validation processes. The integration of Razorpay ensures secure and instant payment confirmation, eliminating delays and reducing fraudulent claims. The QR code-based entry system enables faster and more efficient check-in, minimizing congestion at event venues while preventing duplicate or unauthorized access through single-use validation. Additionally, the analytics dashboard provides real-time insights into registrations, payment status, attendance, and revenue, helping organizers make informed decisions. The system also simplifies report generation through downloadable CSV files, improving documentation and audit processes. Overall, the implementation results in enhanced efficiency, improved security, reduced manual workload, and a better user experience for both students and administrators.

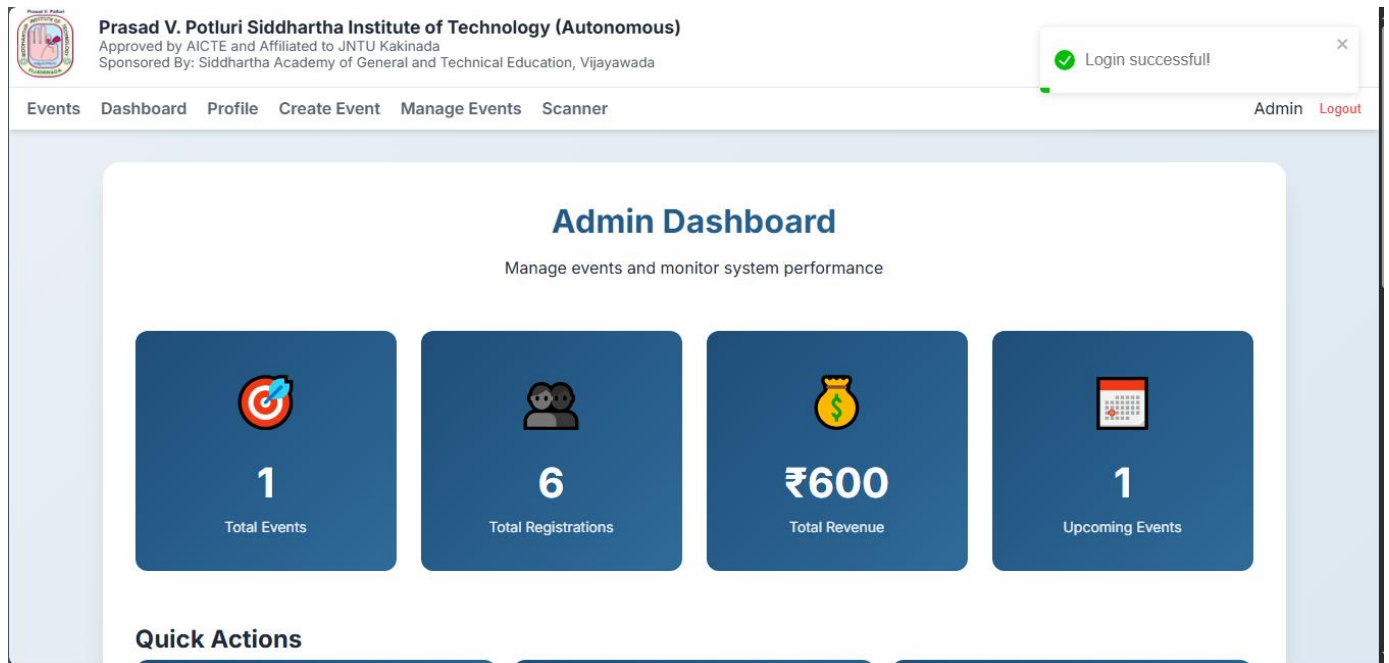


Fig. 2: Admin Dashboard

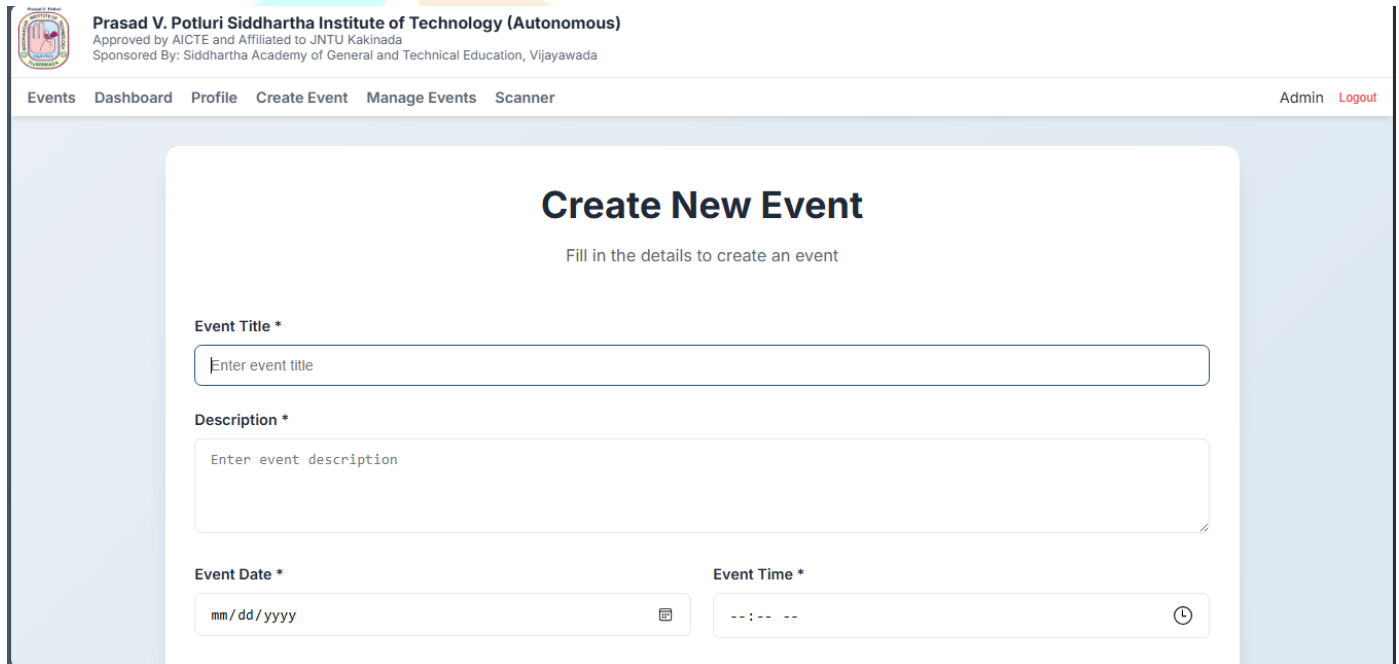


Fig. 3: Admin Creates Events

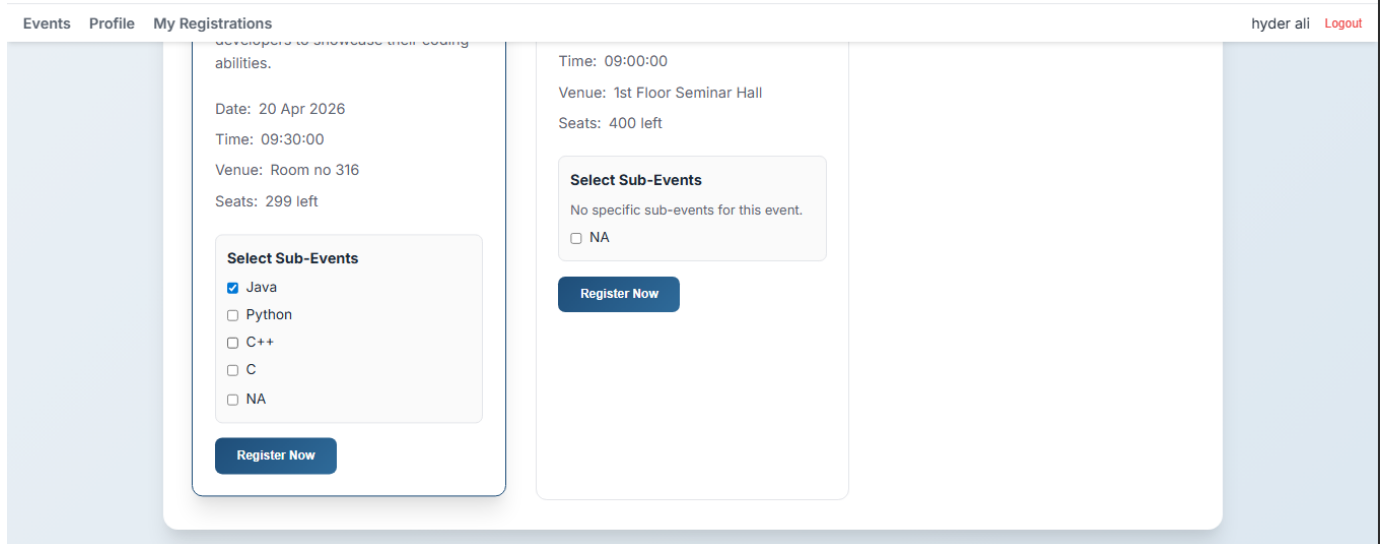


Fig. 4: Student Register Event

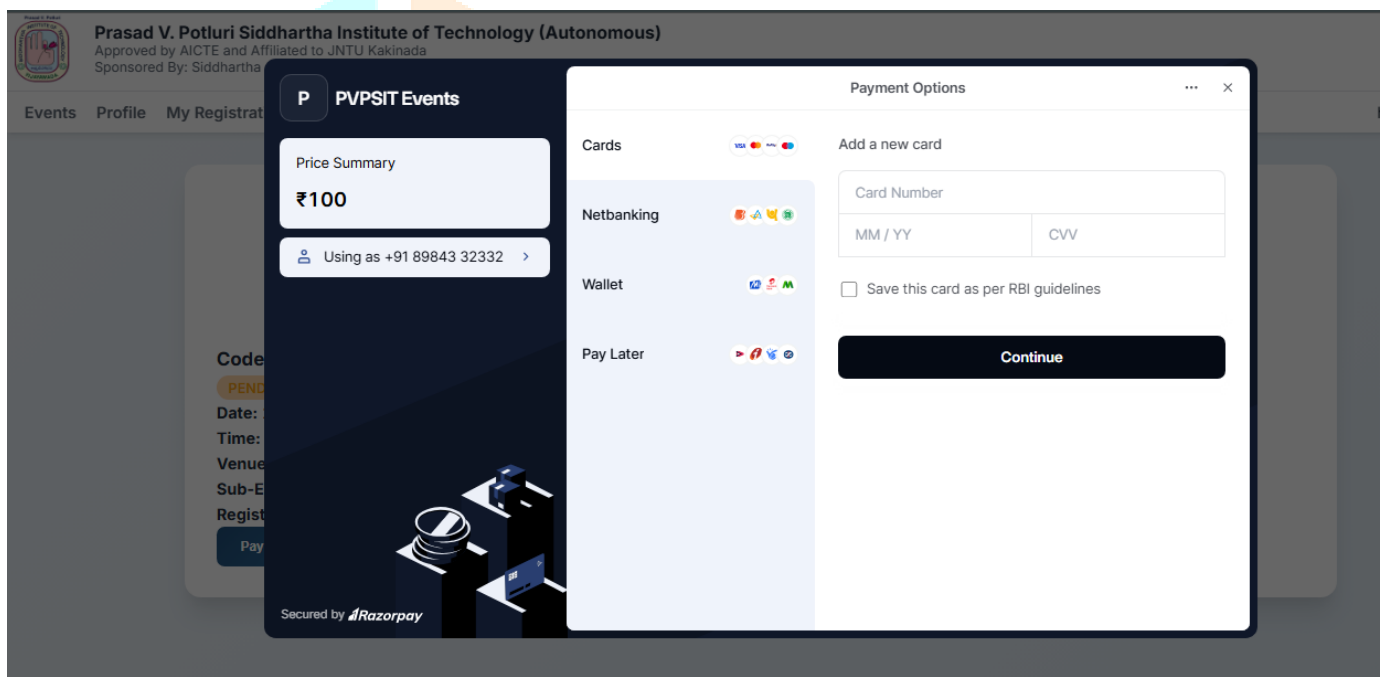


Fig 5: Student Completes Payment

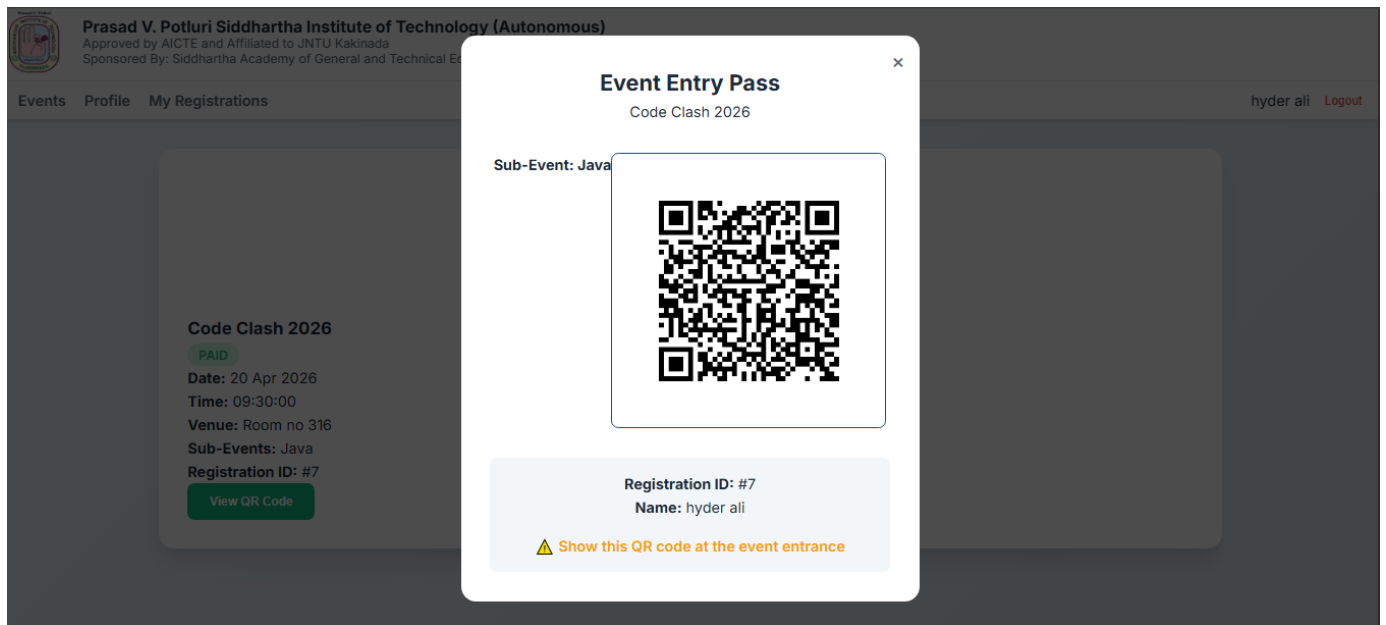


Fig 6: QR Code Generates (Entry pass)

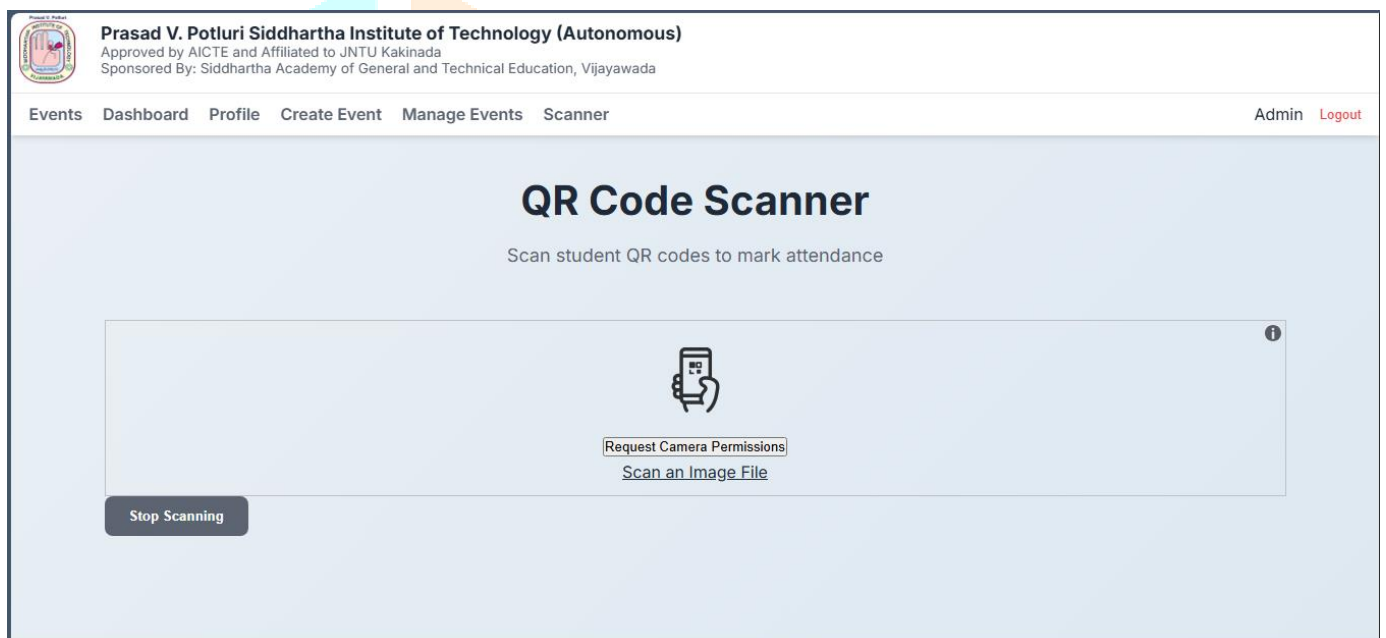


Fig 7: Organizers Scans the student's QR Code & mark attendance as PRESENT

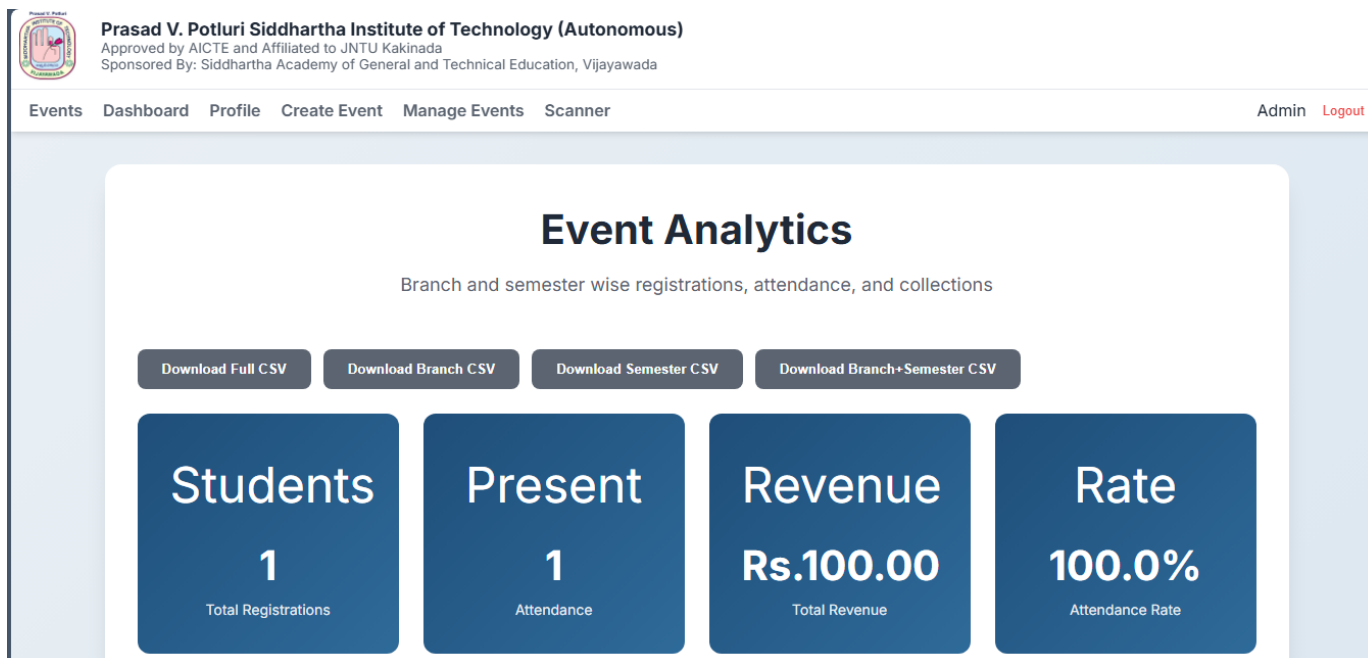


Fig 8.1: Admin’s Event analysis dashboard

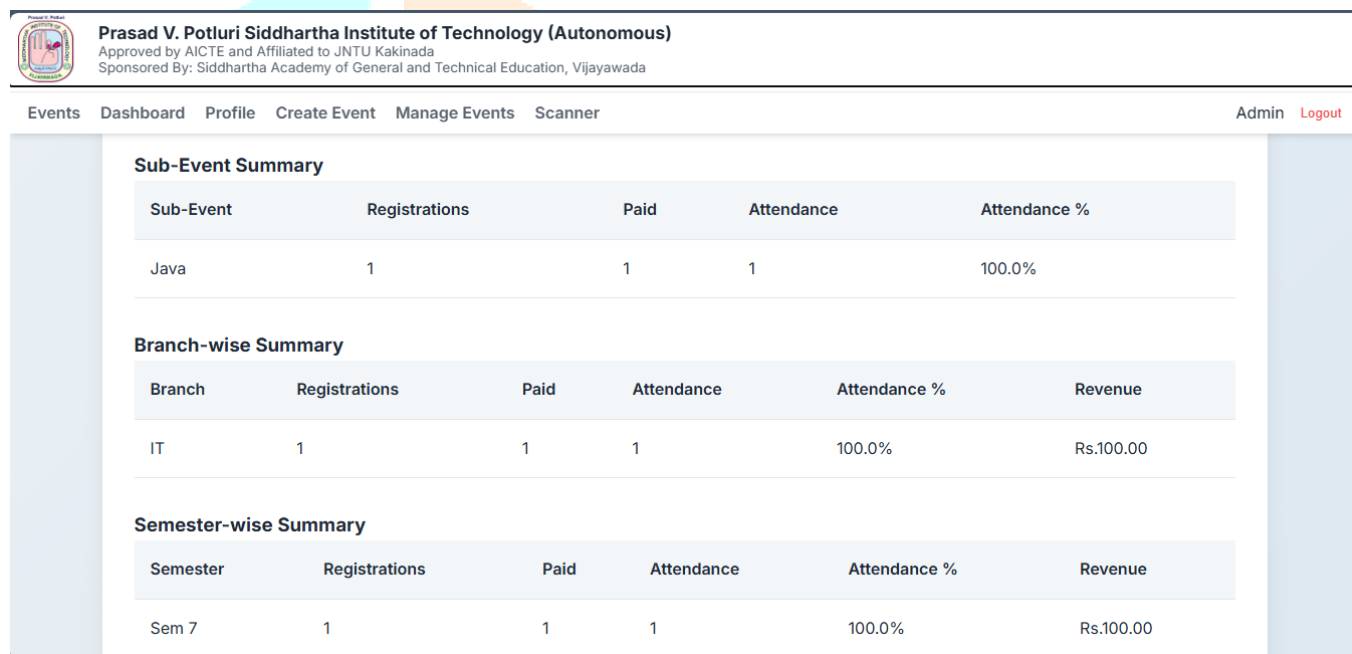


Fig 8.2: Admin’s Event analysis dashboard

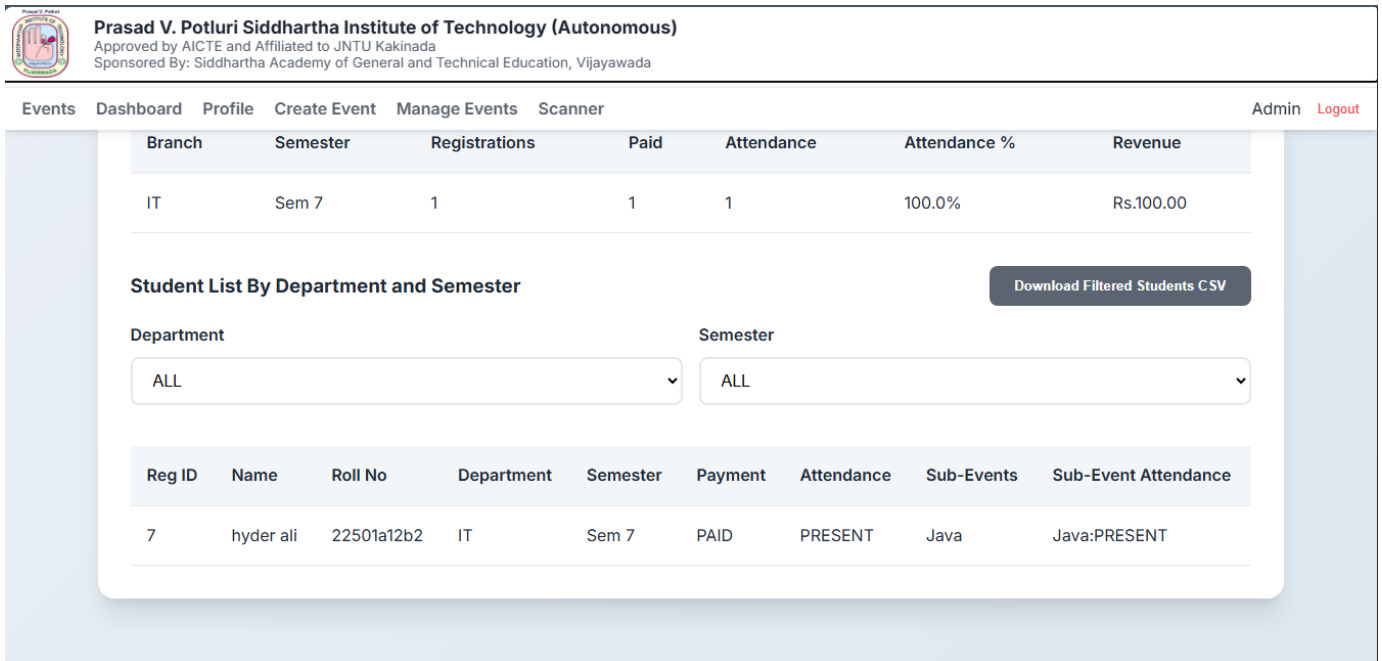


Fig 8.3: Admin's Event analysis dashboard

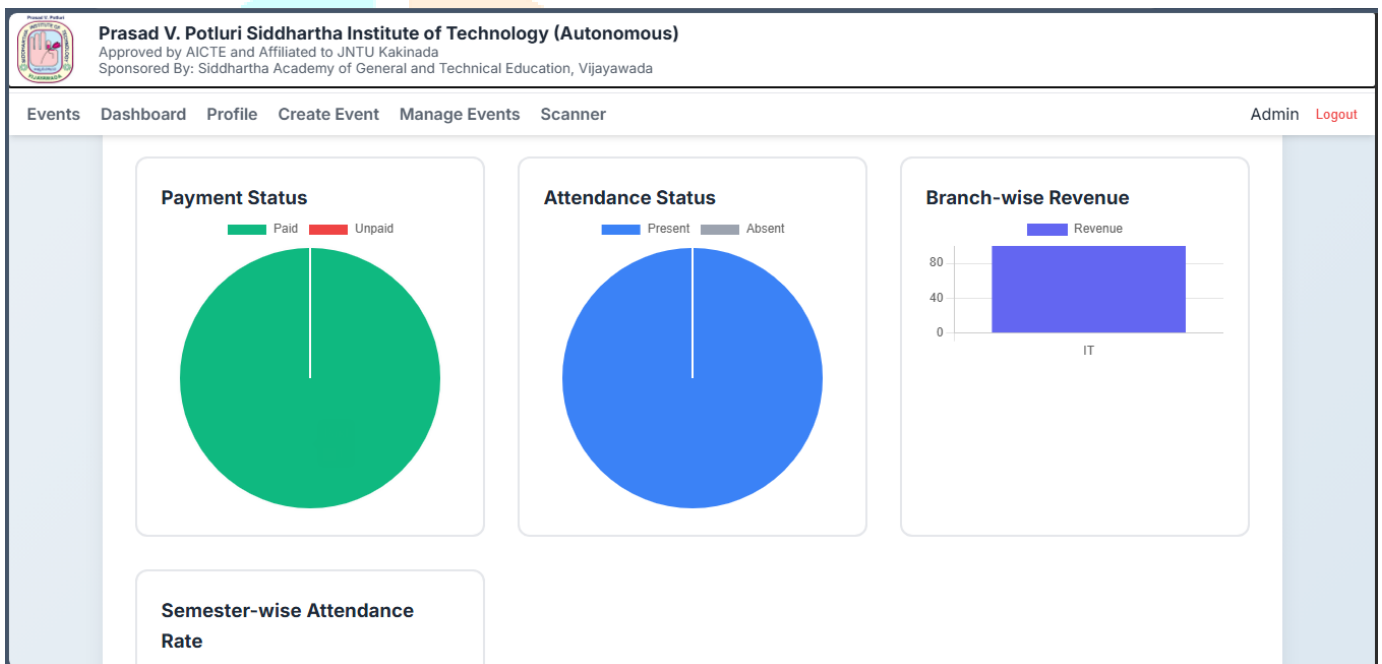


Fig 8.4: Admin's Event analysis dashboard

VI. WEB APPLICATION

The proposed system is developed as a **web-based application** that provides an easy-to-use interface for both students and administrators. The frontend is built using React to ensure a responsive and dynamic user experience, while the backend is developed using Spring Boot to handle business logic, authentication, and API services. The application allows users to access the system from any device with an internet connection, making it convenient and scalable for college-level events. It integrates secure payment processing, QR-based entry validation, and real-time analytics into a single platform, ensuring smooth and efficient event management.

- JWT-based secure authentication with role-based access
- Event creation and management by admin
- Easy student event registration
- Razorpay secure payment integration
- Automatic QR code generation after payment
- QR-based entry validation and attendance marking

- Duplicate entry prevention (single-use QR)
- Real-time analytics dashboard (registration, payment, attendance, revenue)
- CSV report generation
- Responsive design for all devices

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to DR. G.Reshma , Ph.D., Sr. Assistant Professor, for his valuable guidance, constant support, and encouragement throughout the development of this project. His insightful suggestions and continuous motivation played a significant role in the successful completion of this work.

The authors also extend their heartfelt thanks to the Department of Information Technology, Prasad V Potluri Siddhartha Institute of Technology, for providing the necessary infrastructure, resources, and technical support required for carrying out this project. The supportive academic environment and encouragement from the faculty members greatly contributed to enhancing the quality of this work.

Finally, the authors express their sincere appreciation to the management of Prasad V Potluri Siddhartha Institute of Technology for their continuous support and encouragement in successfully completing this project and its presentation.

CONCLUSION

This project modernizes the college event workflow by providing a centralized and secure platform for event registration, payment processing, QR-based entry, attendance marking, and analytics. The key contribution is the integration of Razorpay payment verification, which replaces manual payment confirmation and ensures that only verified payments generate valid QR entry passes. The QR-based entry mechanism improves check-in speed and enforces single-use validation, while the analytics module supports data-driven event planning and documentation. Overall, the system improves efficiency, security, scalability, and transparency for both students and event organizers.

REFERENCES

- [1] Razorpay Developer Documentation – Orders API,Checkout Integration, Payment Signature Verification
- [2] Spring Boot Documentation – REST APIs, Configuration, Transaction Management
- [3] Spring Security Documentation – JWT Authentication and Role-Based Authorization
- [4] React Documentation – Component Architecture and Routing
- [5] ZXing (Zebra Crossing) – QR Code generation libraries and encoding standards
- [6] html5-qrcode Documentation – Browser-based QR scanning using camera stream
- [7] Chart.js Documentation – Data visualization for analytics dashboards
- [8] MySQL Documentation -Relational schema design and query support for reporting