

# College Event Management System

**Khushi Pawar, Rasika Shahane, Roshani Shinde, Harshada Pawar**

**Prof. R. A. Kautkar**

Student, Department of Computer Engineering  
Guru Gobind Singh Polytechnic, Nashik, India  
khushipawar474@gmail.com

**Abstract:** *In modern educational institutions, event management plays a significant role in enhancing student participation and extracurricular development. However, traditional manual systems for managing events often lead to inefficiencies such as data loss, redundancy, miscommunication, and lack of transparency. To address these challenges, a web-based College Event Management System has been developed using the MERN stack, which includes React, Node.js, Express.js, and MongoDB.*

*The proposed system provides a centralized platform where students can view available events and register online. It supports both free and paid events, where students can upload payment proof for verification. The admin module enables administrators to create, update, and delete events, as well as verify registrations and payment screenshots.*

*This system significantly reduces manual workload and improves efficiency by automating event management processes. It ensures data accuracy, transparency, and real-time updates. Additionally, the system provides a user-friendly interface that enhances the overall experience for both students and administrators.*

*The proposed solution is scalable and can be further enhanced by integrating features such as online payment gateways, email notifications, and mobile applications, making it suitable for large-scale deployment in educational institutions..*

**Keywords:** Event Management System, MERN Stack, React, Node.js, MongoDB, Web Application, Admin Dashboard

## I. INTRODUCTION

Event management is an essential activity in educational institutions as it promotes student engagement, teamwork, and skill development [2,3,4]. Colleges regularly organize technical, cultural, and sports events [2]. However, traditional methods of managing these events are often manual and inefficient [2].

Manual systems rely heavily on paperwork and human intervention, which leads to several issues such as data duplication, errors in record-keeping, and difficulty in managing large numbers of participants [2]. Students often face problems in accessing event information, and administrators struggle to track registrations and verify payments [3].

To overcome these limitations, a web-based College Event Management System is proposed [2,3]. This system provides a digital platform that automates the entire event management process [3]. Students can easily browse events, register online, and track their participation status [2]. Administrators can manage events and monitor registrations efficiently [3,4].

The system is developed using the MERN stack [1]. React is used for building a dynamic user interface, Node.js and Express.js handle backend operations, and MongoDB is used for storing data [1,5,6,7]. This architecture ensures scalability, flexibility, and high performance [6,7].

The proposed system improves efficiency by reducing manual work and ensuring accurate data handling [2,4]. It enhances transparency by providing real-time updates and simplifies the overall process of event management [3].



## **II. LITERATURE REVIEW**

Various research studies have been conducted on event management systems, focusing on improving efficiency through digital platforms [2,3,4]. Traditional systems mainly focused on basic registration features without considering scalability, security, and transparency [2].

Earlier systems lacked features such as real-time updates, payment verification, and centralized data storage [2,3]. Many systems were standalone applications, which limited accessibility and usability [3].

Recent advancements in web technologies have led to the development of more sophisticated systems [1]. Web-based applications using frameworks like React and Node.js provide better performance, scalability, and user experience [6,7]. These systems allow real-time interaction and efficient data management [5].

However, many existing systems still lack proper payment verification mechanisms for paid events, which often leads to confusion and fraudulent registrations [2]. Additionally, some systems do not provide role-based access control, which affects security [3].

The proposed system addresses these limitations by integrating modern technologies with essential features such as payment verification, admin control, and status tracking [2,4]. It provides a comprehensive solution that improves transparency and reliability [3].

## **III. PROPOSED SYSTEM**

The proposed College Event Management System is designed to automate and streamline event management processes [2,3]. It is developed using the MERN stack, ensuring high performance and scalability [1,6,7].

The system consists of two main modules:

### **1. Admin Module**

- Add, edit, and delete events [3,4]
- View registered students [2]
- Verify payment screenshots [2]
- Approve or reject registrations [3]

### **2. Student Module**

- View events [2]
- Register for events [2,3]
- Upload payment proof [2]
- Check registration status [3]

The system follows a client-server architecture [1]. The frontend (React) interacts with backend APIs (Node.js + Express.js), which communicate with MongoDB [5,6,7].

Authentication and authorization mechanisms ensure secure access [7]. Role-based access control restricts unauthorized actions [3].

The system ensures transparency by allowing students to track their registration status and enabling admins to verify payments efficiently [2,4].

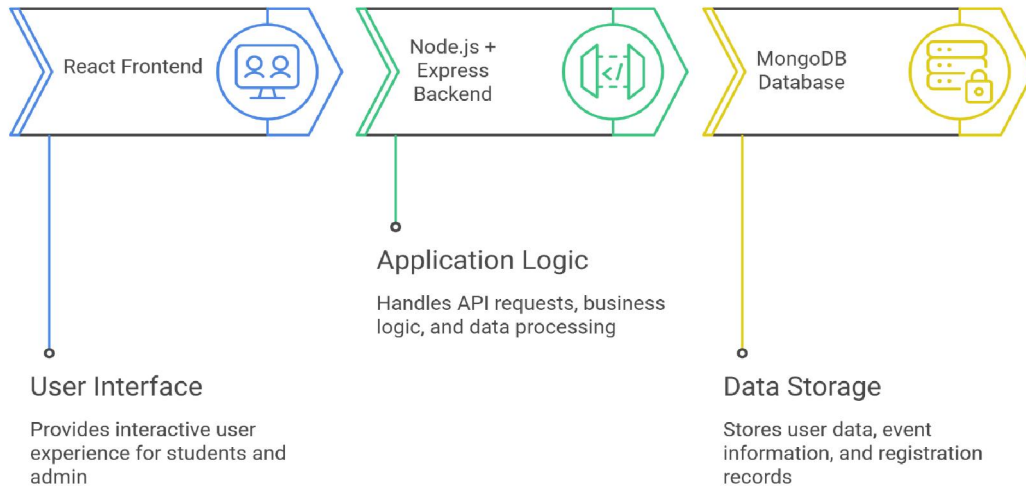


#### IV. SYSTEM ARCHITECTURE

The system follows a **three-tier architecture**[1,6]:

**Architecture Diagram:**

#### Three-Tier System Architecture



**Explanation:**

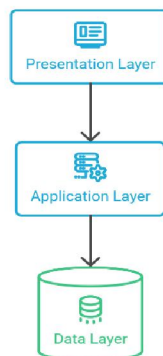
**Presentation Layer (React):** User interface for students and admin [6]

**Application Layer (Node.js):** Handles logic and API requests [7]

**Data Layer (MongoDB):** Stores users, events, registrations [5]

This architecture ensures scalability, modularity, and maintainability [1,6,7].

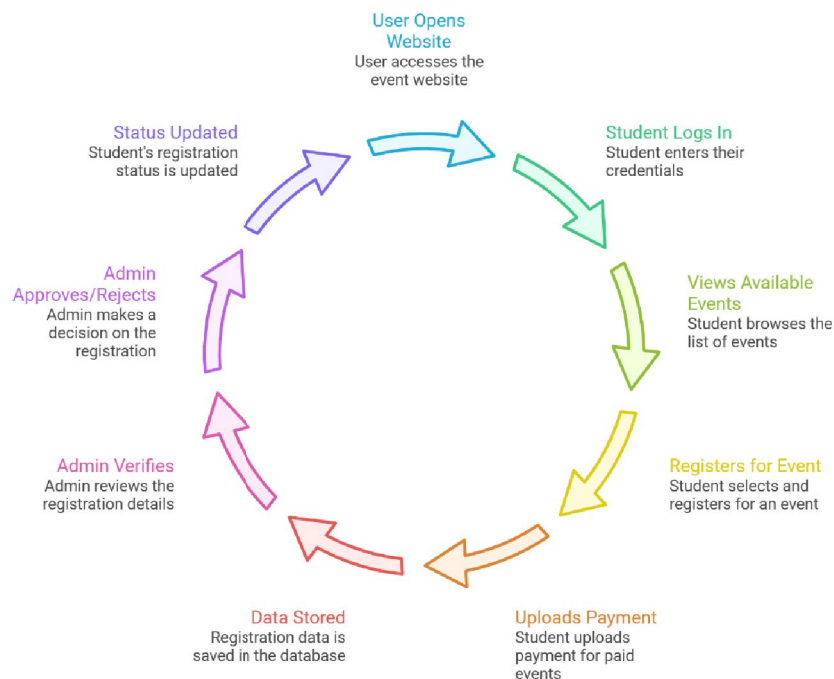
#### System Architecture Flow



### V. SYSTEM WORKING

User opens the website [2]  
 Student logs in [3]  
 Views available events [2]  
 Registers for event [2,3]  
 Uploads payment (if paid event) [2]  
 Data stored in database [5]  
 Admin verifies registration [3,4]  
 Admin approves/rejects [3]  
 Status updated for student [2,3]  
 This workflow ensures smooth communication between users and system [2,4].

### Event Registration Cycle



### VI. DFD (DATA FLOW DIAGRAM)

#### DFD Level 0 (Context Diagram)

[Student] ----> [System] ----> [Database] <---- [Admin] [2,3]

This represents the overall interaction where students and admins communicate with the system, and all data is stored in the database [2,4].



### DFD Level 0 (Context Diagram)



#### DFD Level 1

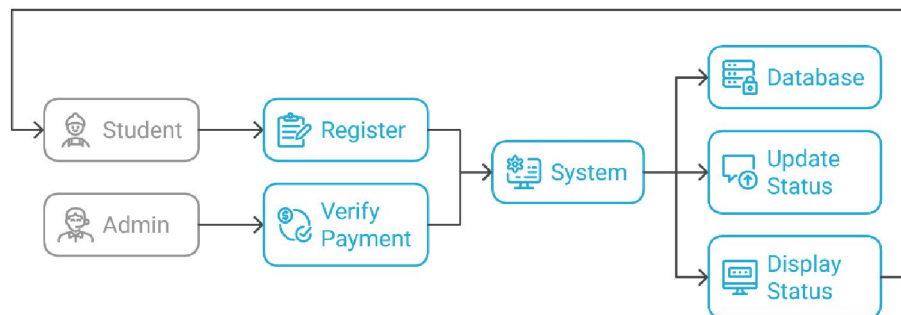
Student → Register → System → Database [2]

Admin → Verify Payment → System → Update Status [3,4]

System → Display Status → Student [2,3]

This level shows detailed data flow between student, admin, and system processes, including registration, payment verification, and status updates [2,4].

### Data Flow Diagram Level 1



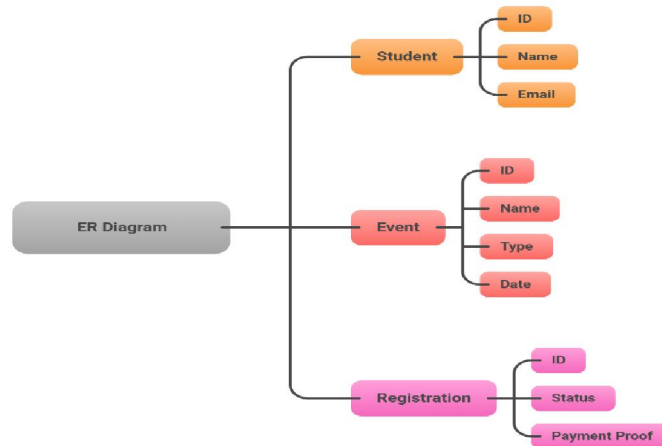
### VII. ER DIAGRAM

#### Entities:

- **Student** (ID, Name, Email) [2]
- **Event** (ID, Name, Type, Date) [2,3]
- **Registration** (ID, Status, Payment Proof) [3,4]

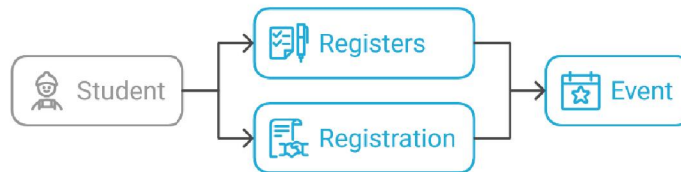


**ER Diagram of Student, Event, and Registration**



**Diagram :**

**Student Event Registration Process**



**VIII. ADVANTAGES**

- Reduces manual work [2]
- Improves transparency [3]
- Centralized data system [5]
- Easy access for students [2]
- Efficient admin control [3,4]
- Scalable system [1,6]

**IX. FUTURE SCOPE**

- Online payment gateway [2]
- Email notifications [6]
- Mobile application [6,7]
- QR-based entry system [4]
- Analytics dashboard [4,5]



### **X. CONCLUSION**

The College Event Management System provides an efficient solution for managing events in educational institutions [2,3]. It automates the registration process, improves transparency, and reduces manual workload [2,4]. The use of MERN stack ensures scalability and performance [1,6,7]. The system enhances user experience and provides a structured approach to event management [3].

### **ACKNOWLEDGMENT**

The authors would like to express their sincere gratitude to all those who contributed to the successful completion of this project. We are especially thankful to our project guide, **Prof. R. A. Kautkar**, for his valuable guidance, continuous support, and encouragement throughout the development of this system. His insights and suggestions greatly helped us in improving the quality and effectiveness of our project.

We would also like to thank the faculty members of the Department of Computer Engineering at **Guru Gobind Singh Polytechnic, Nashik**, for providing us with the necessary resources and knowledge required for completing this work.

We are grateful to our institution for offering a supportive environment and infrastructure that enabled us to successfully implement this project. Lastly, we extend our appreciation to our friends and family members for their motivation and support throughout the project duration.

### **REFERENCES**

- [1]. Bawane M, Gawande I, Joshi V, Nikam R, Bachwani SA. A Review on Technologies used in MERN stack. *Int. J. Res. Appl. Sci. Eng. Technol.* 2022 Jan;10(1):479-88.
- [2]. Bhulakshmi D, Naveen U, Suresh V, Atmakuri L, Reddy TR. Web application of college event management. In *AIP Conference Proceedings 2024 Feb 13 (Vol. 2742, No. 1, p. 020056)*. AIP Publishing LLC.
- [3]. Shara C, Sarala V. Online Event Management System. *International Journal of Management Research and Reviews*. 2025 Apr 1;15(2s):330-2.
- [4]. Lakshmi MB, Anand AV, Akshit RS, Akshara M, Varma KB, Pradeep LC. University Event Management System using Data Visualization. In *2025 International Conference on Next Generation of Green Information and Emerging Technologies (GIET) 2025 Aug 8 (pp. 1-13)*. IEEE.
- [5]. Chauhan A. A review on various aspects of MongoDB databases. *International Journal of Engineering Research & Technology (IJERT)*. 2019 May;8(05):90-2.
- [6]. Hoque S. *Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js*. Packt Publishing Ltd; 2020 Apr 17.
- [7]. Mardan A. *Pro express.js: Master express.js: The node.js framework for your web development*. Apress; 2014 Dec 24.

