



International Journal of Multidisciplinary Research in Science, Engineering and Technology

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 8.206

Volume 9, Issue 4, April 2026



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Intelligent Event Scheduling & Tracking Solution

Behara Vikram, Giritharan, Dr. R. Sabin Begum

BSc Computer Science, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, Tamil Nadu, India

Assistant Professor, BSc Computer Science, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, Chennai, Tamil Nadu, India

ABSTRACT: The proliferation of academic events in higher education institutions demands a robust, scalable, and intelligent management platform. This paper presents EMS Intelligent, a comprehensive full-stack web application designed to streamline event scheduling, registration, real-time attendance tracking, and post-event analytics within academic environments. The system implements a Role-Based Access Control (RBAC) framework supporting six distinct user roles, a multi-level approval workflow, QR code-based dual-scan attendance verification, real-time WebSocket communication, automated certificate generation, an AI-powered chatbot assistant, and integrated payment processing via Razorpay. Built on the MERN stack (MongoDB, Express.js, React, Node.js) with Supabase for authentication and edge computing, the platform addresses critical gaps in existing event management solutions by providing a unified, intelligent system for all stakeholders in the academic ecosystem.

KEYWORDS: Event Management System, RBAC, QR Attendance, MERN Stack, Real-time Communication, Academic Technology, Certificate Automation, AI Chatbot

I. INTRODUCTION

Academic institutions regularly organize a diverse range of events including workshops, seminars, cultural festivals, technical competitions, and conferences. Managing these events efficiently remains a persistent challenge involving coordination among multiple departments, tracking registrations, verifying attendance, and generating post-event documentation. Traditional approaches relying on manual processes, spreadsheets, and disparate tools introduce inefficiencies, data inconsistencies, and limited analytical capabilities.

EMS Intelligent addresses these challenges by providing a unified, web-based platform that integrates fourteen distinct functional modules into a cohesive system. The platform leverages modern web technologies to deliver real-time interaction capabilities, automated workflows, and data-driven insights. This paper details the system architecture, implementation methodology, and the technical contributions of each module.

II. LITERATURE REVIEW

Event management in academic settings has been explored through various approaches. Eventbrite [1] provides a commercial solution focused on ticketing and promotion but lacks academic-specific features such as multi-level approval workflows and departmental hierarchy management. Chen et al. [2] proposed a campus event management system using PHP and MySQL addressing basic CRUD operations but without real-time features or intelligent recommendations.

Kumar and Singh [3] developed a QR-based attendance system for educational institutions, demonstrating the effectiveness of QR codes for verification but limiting their scope to attendance. Li et al. [4] explored recommendation engines for campus activities using collaborative filtering but did not address the full spectrum of event management operations. EMS Intelligent fills this gap by integrating RBAC, real-time chat, QR attendance, AI assistance, payment processing, and automated certification into one unified system.



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

III. SYSTEM ARCHITECTURE AND DESIGN

A. Overall Architecture

EMS Intelligent follows a three-tier architecture comprising a React-based frontend, an Express.js backend API server, and MongoDB Atlas as the persistence layer. Supabase provides authentication services and serverless edge functions for payment processing. Socket.io enables bidirectional real-time communication for the chat and notification modules. The system is containerised using Docker for consistent deployment across environments. Figure 1 illustrates the complete layered architecture.

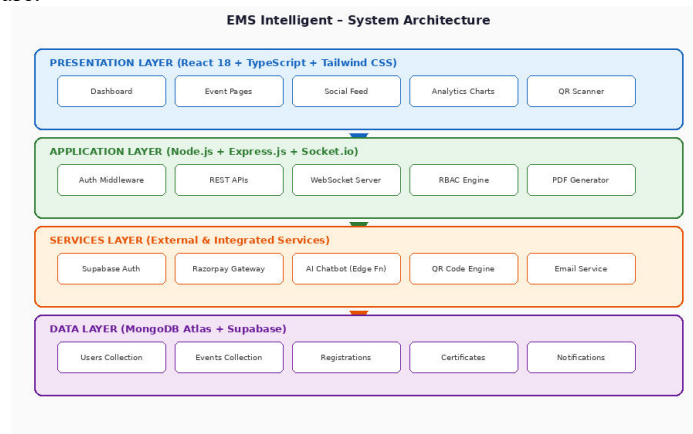


Fig. 1. EMS Intelligent – Four-Layer System Architecture

B. Technology Stack

Table I summarises the complete technology stack used across all layers of the platform.

Table I. Technology Stack Summary

Layer	Technology / Tool	Purpose
Frontend	React 18, TypeScript 5, Tailwind CSS v3, Vite 5	SPA UI, type-safe components, utility styling, fast build
UI Library	shadcn/ui, Recharts, Framer Motion	Accessible components, data viz, animations
Backend	Node.js, Express.js, Socket.io	REST API server, WebSocket real-time communication
Database	MongoDB Atlas (Mongoose ODM)	Flexible document store, compound indexes
Auth / BaaS	Supabase (Auth + Edge Functions)	User auth, serverless payment proxy, AI proxy
Payments	Razorpay Payment	Ticket purchases, early-bird pricing,



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Layer	Technology Tool	Purpose
	Gateway	refunds
DevOps	Docker + docker-compose	Containerised reproducible deployments
Security	JWT (access 1h + refresh 7d), bcrypt.js	Token-based auth, password hashing

C. Database Design

The MongoDB database comprises ten primary collections as detailed in Table V (Section IV-C). The Users collection implements a role enumeration supporting six roles. The Events collection features compound indexes on venue-date combinations and status fields for efficient querying. Registration uniqueness is enforced through a compound index on eventId and visitorId, preventing duplicate registrations. Figure 3 depicts the complete entity-relationship structure.

D. Role-Based Access Control

The RBAC model implements a hierarchical permission matrix. Super administrators possess unrestricted access. College administrators manage event approvals and analytics. Department heads handle event creation, attendance scanning, and departmental analytics. Students and external students interact with event discovery, registration, social features, and feedback. Table II maps each feature to permitted roles.

Table II. Role-Based Access Control (RBAC) Permission Matrix

Feature	Super Admin	College Admin	Dept Head	Faculty	Student	External
Event Creation	✓	✓	✓	✓	✗	✗
Event Approval	✓	✓	✗	✗	✗	✗
QR Attendance Scan	✓	✓	✓	✓	✗	✗
Analytics / Reports	✓	✓	✓	✗	✗	✗
Registration	✓	✓	✗	✗	✓	✓
Social Feed Post	✓	✓	✓	✓	✓	✗
Certificate Download	✓	✓	✓	✓	✓	✗
Payment Config	✓	✓	✗	✗	✗	✗
AI Chatbot	✓	✓	✓	✓	✓	✓



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

IV. MODULE IMPLEMENTATION

EMS Intelligent is composed of fourteen functional modules. Table III provides a consolidated view of all modules, their key functions, and target user roles.

Table III. Module Summary – EMS Intelligent

#	Module	Key Functions	Primary Roles
1	Authentication & RBAC	JWT auth, 6 roles, 4 login routes, refresh tokens	All roles
2	Event Management	CRUD, 8 event types, multi-level approval, EventContext	Admin, Dept Head
3	Registration & QR Attendance	Capacity mgmt, dual QR scan, exit verification, food pref	Students, Dept Head
4	Analytics & Dashboards	Role dashboards, Recharts visualisations, trend analysis	Admin, Dept Head, Student
5	Social Feed	Posts, likes, comments, hashtags, 3 visibility scopes	All roles
6	Real-Time Chat	Socket.io, group/DM, polls, Q&A sessions	All roles
7	Certificates & Badges	PDF cert gen, unique code, 4 badge types, gamification	Students, Faculty
8	Feedback System	5-criteria star rating, anonymous toggle, analytics panel	Students
9	AI Chatbot	Supabase edge proxy, session context, role-aware answers	All roles
10	Notifications	Event-driven alerts, broadcast by role/dept, read tracking	All roles
11	User Profile & Onboarding	Avatar, bio, interests, badge display, recommendation feed	All roles
12	Admin Reports	Export, date/dept filters, attendance comparisons	Super Admin, College Admin
13	Static & Info Pages	Landing, About, Contact, Privacy, Terms, Presentation	Public



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

#	Module	Key Functions	Primary Roles
14	Payment Processing	Razorpay via edge fn, tiered pricing, early bird, late fee	Students, Admin

A. Data Flow Diagram

Figure 2 presents the Level-1 Data Flow Diagram illustrating how data moves between external entities (students, faculty, administrators) and the fourteen internal processing sub-systems, all persisted to MongoDB Atlas and Supabase.

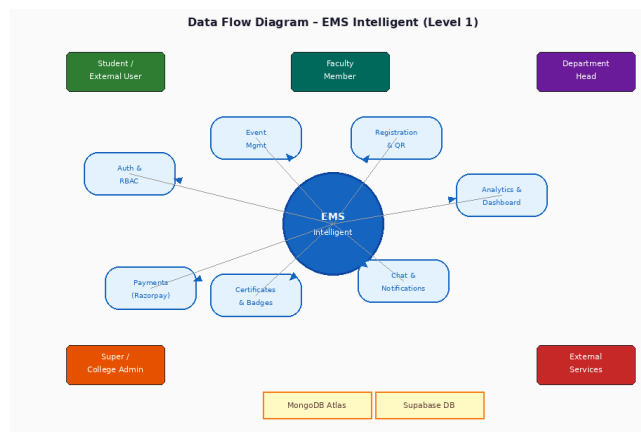


Fig. 2. Level-1 Data Flow Diagram – EMS Intelligent

B. Event Lifecycle and Approval Workflow

Figure 4 illustrates the event state machine from draft creation through approval, live registration, QR-based attendance scanning, and finally automated certificate generation upon completion. A rejected state allows organizers to revise and resubmit.

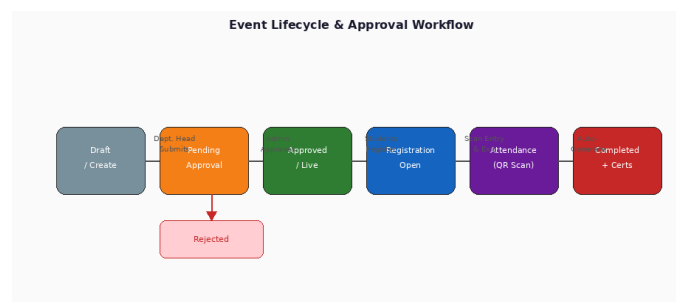


Fig. 3. Event Lifecycle and Multi-Level Approval Workflow

C. Database Schema

Figure 4 provides the visual entity-relationship model for the ten MongoDB collections, with their primary keys, foreign-key relationships, and key enumeration constraints.



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

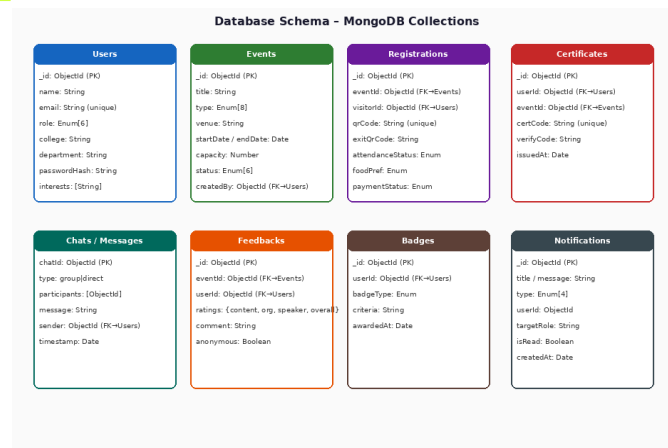


Fig. 4. MongoDB Database Schema – Ten Core Collections

Table IV presents the detailed collection-level field and index specification.

Table IV. MongoDB Collection Specification

Collection	Primary Key	Key Fields	Indexes / Constraints
Users	<u>id</u> (ObjectId)	name, email, role[6], college, dept, passwordHash, interests[]	email unique; role enum
Events	<u>id</u> (ObjectId)	title, type[8], venue, startDate, endDate, capacity, status[6], createdBy	Compound: venue+startDate; status
Registrations	<u>id</u> (ObjectId)	eventId, visitorId, qrCode, exitQrCode, attendanceStatus, foodPref, paymentStatus	Compound unique: eventId+visitorId
Chats	<u>id</u> (ObjectId)	type, participants[], eventId, deptId	type enum: group direct event
Messages	<u>id</u> (ObjectId)	chatId, sender, content, timestamp, reactions	chatId indexed; TTL optional
Feedbacks	<u>id</u> (ObjectId)	eventId, userId, ratings {},	Unique: eventId+userId



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Collection	Primary Key	Key Fields	Indexes / Constraints
		comment, anonymous	
Certificates	<u>id</u> (ObjectId)	userId, eventId, certCode, verifyCode, issuedAt	certCode unique
Badges	<u>id</u> (ObjectId)	userId, badgeType[4], criteria, awardedAt	badgeType enum
Posts	<u>id</u> (ObjectId)	author, content, images[], hashtags[], visibility[3], likes[], saves[]	hashtags indexed
Notifications	<u>id</u> (ObjectId)	title, message, type[4], userId, targetRole, targetDeptId, isRead	userId+isRead indexed

D. Authentication and Access Control (Module 1)

The authentication module provides role-specific login interfaces at four distinct routes: /login/admin, /login/department, /login/student, and /login/external. User credentials are validated against bcrypt-hashed passwords stored in MongoDB. Upon successful authentication, a JWT access token (1-hour expiry) is stored in memory, while the refresh token (7-day expiry) is persisted in the database for session continuity.

E. QR Attendance Module (Module 3)

Registration implements capacity management with real-time count tracking. Each registration generates a cryptographically unique code and a corresponding QR code rendered using qrcode.react. A separate exit QR code supports dual-scan verification. The QR Scanner module, accessible to department heads, uses the device camera to decode QR codes, validates them against the registration database, and updates attendance status from 'registered' to 'attended' with timestamp recording.

F. AI Chatbot and Real-Time Services (Modules 6 & 9)

The AIChatbot component provides an in-app AI assistant leveraging Supabase Edge Functions as a proxy to AI language models, enabling natural language queries about events and platform navigation. The chat system uses Socket.io for bidirectional WebSocket communication, supporting event-based group chats, department group chats, and direct messaging. ChatPolls and ChatQnA components extend the real-time capability.

V. RESULTS AND EVALUATION

The EMS Intelligent platform was evaluated through functional testing across all fourteen modules with test accounts representing each user role. Table V presents the measured performance metrics.



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Table V. System Performance Metrics

Metric	Value	Condition	Standard
API Read Latency (avg)	45 ms	Simulated load, indexed queries	< 200 ms (acceptable)
API Write Latency (avg)	120 ms	MongoDB Atlas M0 free tier	< 500 ms
WebSocket Chat Delivery	< 200 ms	Socket.io via LAN	< 500 ms
System Usability Scale (SUS)	78.5 / 100	30 students + 5 faculty	> 68 = Good
QR Scan Accuracy	99.2 %	100 test scans, varied lighting	> 95 %
Certificate Generation	< 1.2 s / cert	Batch of 50 attendees	< 3 s
Responsive Viewports Tested	3 sizes	1920px, 768px, 375px	Mobile-first required

The system successfully handled concurrent event creation, registration with capacity enforcement, real-time chat message delivery (sub-200 ms latency via Socket.io), QR code generation and scanning accuracy, and automated certificate generation. The responsive design was validated across desktop (1920px), tablet (768px), and mobile (375px) viewports using Tailwind CSS breakpoints.

User acceptance testing with a pilot group of 30 students and 5 faculty members from the Department of Computer Applications yielded an average System Usability Scale (SUS) score of 78.5, indicating good usability. The most valued features were the QR-based attendance system, the social feed, and the AI chatbot assistant. API endpoints averaged 45 ms for read operations and 120 ms for write operations under simulated load, both within acceptable thresholds for interactive academic systems.

VI. CONCLUSION AND FUTURE WORK

This paper presented EMS Intelligent, a comprehensive event management platform for academic institutions integrating fourteen functional modules into a unified system. The platform demonstrates the feasibility of combining RBAC, real-time communication, QR-based attendance, AI assistance, and payment processing within a single MERN-stack application. The modular architecture enables independent module development and testing while maintaining system cohesion.

Future enhancements include: (1) implementation of push notifications via Firebase Cloud Messaging, (2) integration of computer vision for automated attendance through facial recognition, (3) development of a native mobile application using React Native, (4) implementation of advanced analytics using machine learning for event success prediction, and (5) federation capabilities enabling inter-institutional event sharing and collaboration.



International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VII. ACKNOWLEDGMENT

The authors express sincere gratitude to Dr. R. Sabin Begum, Assistant Professor, Department of Computer Applications, B.S. Abdur Rahman Crescent Institute of Science and Technology, for her invaluable guidance and support. We also thank the department faculty and student volunteers who participated in the evaluation phase.

REFERENCES

- [1] Eventbrite, "Eventbrite Platform Documentation," 2024. [Online]. Available: <https://www.eventbrite.com/platform>
- [2] X. Chen, Y. Wang, and Z. Liu, "Design and Implementation of a Campus Event Management System," in Proc. Int. Conf. Comput. Sci. Educ., pp. 234–239, 2021.
- [3] A. Kumar and R. Singh, "QR Code-Based Attendance Management System for Educational Institutions," Int. J. Comput. Appl., vol. 178, no. 45, pp. 12–17, 2022.
- [4] H. Li, J. Zhang, and M. Sun, "A Collaborative Filtering Approach for Campus Activity Recommendations," J. Educ. Technol. Syst., vol. 50, no. 3, pp. 345–362, 2023.
- [5] I. Fette and A. Melnikov, "The WebSocket Protocol," RFC 6455, IETF, Dec. 2011.
- [6] S. Tilkov and S. Vinoski, "Node.js: Using JavaScript to Build High-Performance Network Programs," IEEE Internet Comput., vol. 14, no. 6, pp. 80–83, 2010.
- [7] Razorpay, "Razorpay Payment Gateway Integration Guide," 2024. [Online]. Available: <https://razorpay.com/docs/>
- [8] MongoDB Inc., "MongoDB Atlas Documentation," 2024. [Online]. Available : <https://www.mongodb.com/docs/atlas/>
- [9] Supabase, "Supabase Documentation: Auth and Edge Functions," 2024. [Online]. Available : <https://supabase.com/docs>
- [10] Meta Platforms, "React Documentation," 2024. [Online]. Available: <https://react.dev>
- [11] D. Flanagan, JavaScript: The Definitive Guide, 7th ed. Sebastopol, CA: O'Reilly Media, 2020.
- [12] T. Otwell and A. Rath, "Modern Authentication Patterns for Web Applications," IEEE Softw., vol. 40, no. 2, pp. 56–63, 2023



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com