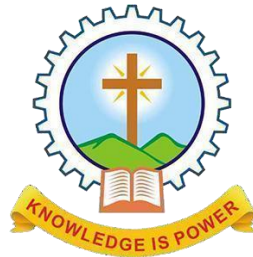


MAR ATHANASIUS COLLEGE OF ENGINEERING
(Affiliated to APJ Abdul Kalam Technological University, TVM)
KOTHAMANGALAM



INITIAL REPORT
ON
CAMPUS CONNECT
EVENT REGISTRATION SYSTEM

Submitted By:

Akhil Rock Babu

MAC23MCA-2010

Faculty Guide:

Prof. Biju Skaria

Project Coordinator:

Prof. Sonia Abraham

ABOUT THE PROJECT

CampusConnect - Event Registration System

The CampusConnect - Event Registration System is a web-based application developed to streamline and automate the management of events hosted by colleges or educational institutions. This project aims to replace traditional, manual methods of event registration and coordination with a centralized and efficient digital solution. Events conducted as part of tech fests, cultural programs, and other academic activities often face challenges such as managing participant registrations, organizing event schedules, and ensuring seamless communication. The need for an interactive and user-friendly system is essential for enhancing the overall experience of event management.

The existing system for event management in most institutions relies on basic websites, spreadsheets, or Google Forms. These static websites primarily serve as informational portals, offering only limited functionalities such as event details, contact information, and basic registration forms. They lack the capability of a dynamic form builder, which is essential for tailoring forms to the specific requirements of different events. This limitation makes it challenging to collect varied participant information for different types of events. Moreover, such traditional systems often do not provide options for customization in the interface, making it difficult for institutions to create a personalized and professional experience. Communication with participants in such systems is often fragmented, leading to delays and inefficiencies in managing event updates and coordination.

The Proposed System is a comprehensive web-based platform designed to streamline event management in educational institutions. Catering to administrators, organizers, and participants, it offers features like online registration, profile management, and event customization, making it highly adaptable to institutional needs. A key highlight is the Dynamic Form Builder, allowing organizers to create tailored forms to collect event-specific requirements, addressing limitations of traditional systems. The interface is customizable, enabling institutions to align the platform with their branding and functional needs. Participants benefit from simplified registration, profile updates, and access to event history, while organizers can efficiently manage schedules, categories, and participants. Secure fee payment integration, real-time communication via email or SMS, and ticket generation with QR codes ensure seamless event operations.

The system is built using modern web technologies, including HTML, CSS, and JavaScript for an interactive and responsive frontend, PHP for robust server-side logic, and MongoDB for efficient and flexible data storage and management.

In summary, the CampusConnect - Event Registration System revolutionizes event management by addressing the limitations of traditional systems. Its dynamic and customizable features, combined with modern technology, make it an indispensable tool for educational institutions, enhancing the experience for both organizers and participants.

BUSINESS RULES

Event Registration:

- Participants must register for events before the specified registration deadline.
- A participant can register for multiple events but must complete the registration process for each event individually.
- Registrations are processed on a first-come, first-serve basis until event capacity is reached.

Event Payment:

- Participants must make event payments within the designated payment window.
- Payment must be confirmed before participants can receive event tickets or access event materials.

Ticketing and Entry:

- Each participant will receive a ticket.
- Not having event tickets will be denied entry.

Event Updates and Notifications:

- Notifications for any schedule changes or important updates must be sent to participants through the platform (email).

Admin Approval:

- Organizers can only create and manage events after being approved by institutional admins.

Participant Data Privacy:

- All personal data collected during registration must be securely stored and not shared with third parties without the participant's consent.
- Participants have the right to update their registration information at any time.

LIST OF SCENARIOS

1. Website Customization

- Admins can customize website interface by choosing preferred font, colour.
- Admins can customize background images and banners by uploading the required resources.
- Admins can customize Interface texts by filling the details.

2. User Registration and Login Scenarios

- An organizer registers on the platform to create and manage events.
- An admin reviews and approves the organizer's account.
- Organizers receives email for approval or rejection with reason.
- Organizers can now login to start creating and manage events.

3. Event Management Scenarios

- An organizer creates a new event by providing event details (name, date, venue, description, ticket price, etc.).
- An organizer uses the jQuery form builder to add specific fields.
- An organizer updates event details after creation.
- An organizer deletes a cancelled event.
- An organizer adds co-organizers to assist in managing events.

4. Registration and Payment Scenarios

- A participant views a list of available events.
- A participant selects an event and completes the registration process.
- A participant uploads required documents for the event during registration.
- A participant makes an online payment for the event and receives a receipt.

5. Ticket Generation and Verification Scenarios

- After successful registration and payment, organizers are assigned to approve/reject the registration.
- After approval participants receives tickets for the events.
- An organizer verifies participant tickets at the event venue for entry verification.

6. Communication and Feedback Scenarios

- A participant provides feedback or reviews on the event after attending.
- An admin addresses participant feedbacks and ensures resolution.

DESCRIPTION OF MODULES

Organizer Management Module:

- Organizers can self-register on the platform with necessary details.
- Organizers' registrations are subject to approval by institutional admins. Once approved, they can access event creation and management features.
- Login credentials are generated during registration, and organizers can login using that credentials.
- Organizers can then add Co-organizers for collaboration.

Event Management Module:

- Approved Organizers can create, update, and manage event details, including schedules, categories, and venue information.
- Utilizes the Dynamic Form Builder to create customized registration forms for specific events.
- Enables event organizers to manage participant registrations and generate event reports.
- Enables organizers to send updates to participants registered for events via email.

Participants Management Module:

- Participants can register for single or multiple events via an intuitive interface.
- Participants are notified about event-related updates or changes in real time.
- Organizers can monitor and manage participant registrations and track registration statuses.
- Participants can view fee details and make secure online payments through integrated payment methods.

Customization Module:

- Institutional Admins can customize the platform interface to align with their branding and functional needs.

Feedback Management Module:

- Participants can provide feedback about events, services, or issues encountered.

ACTORS AND THEIR ROLES

The CampusConnect - Event Registration System involves multiple actors, each contributing to the efficient management and seamless execution of events. These actors include Institution Administrators, Organizers, Co-Organizers, and Participants.

Institution Administrator

- Oversees the entire system and manages user access for organizers.
- Sets up institution-specific branding and customization options.
- Manages event categories and institutional preferences.
- Monitors event schedules, registrations, and overall system performance.
- Generates detailed reports on event participation, ticket sales, and system analytics.

Organizer

- Creates and manages event details, including schedules, categories, and venues.
- Utilizes the Dynamic Form Builder to create tailored registration forms for specific events.
- Manages participant registrations and assigns roles to co-organizers.
- Sends notifications or updates to participants via email.
- Tracks registration payments and generates invoices for fee-based events.
- Oversees ticket generation and distribution with QR codes for entry verification.

Co-Organizer

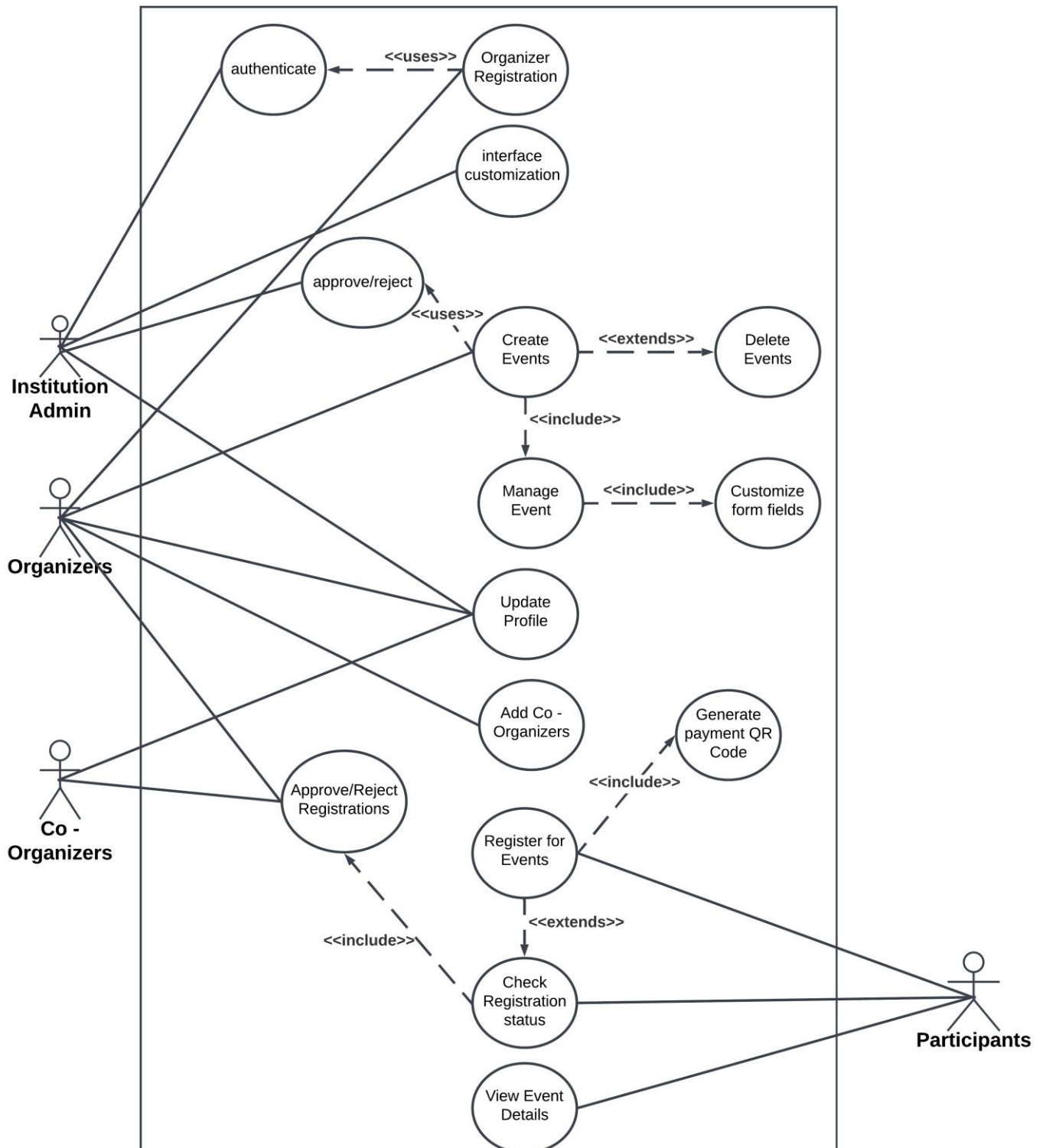
- Assists organizers in managing event details and registrations.
- Handles participant queries and provides necessary support.
- Manages on-site event logistics, including attendance tracking and entry verification.
- Monitors ticket verification and ensures smooth participant entry at venues.

Participant

- Registers for single or multiple events through a user-friendly interface.
- Updates and manages their profile details.
- Receives event-related updates via email.
- Pays event fees securely through integrated payment methods.
- Accesses tickets with QR codes and tracks their event history.
- Provides feedback on events and services to improve the overall experience

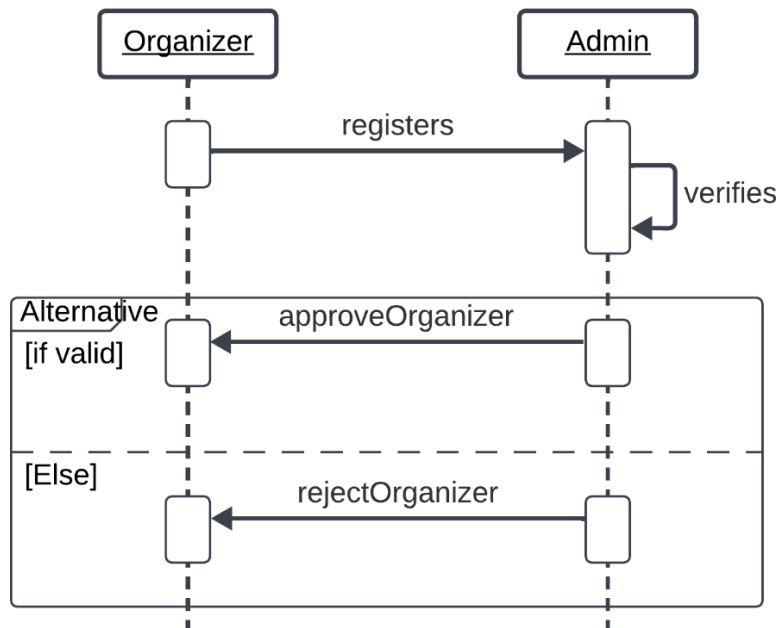
UML DIAGRAMS

USE CASE DIAGRAM

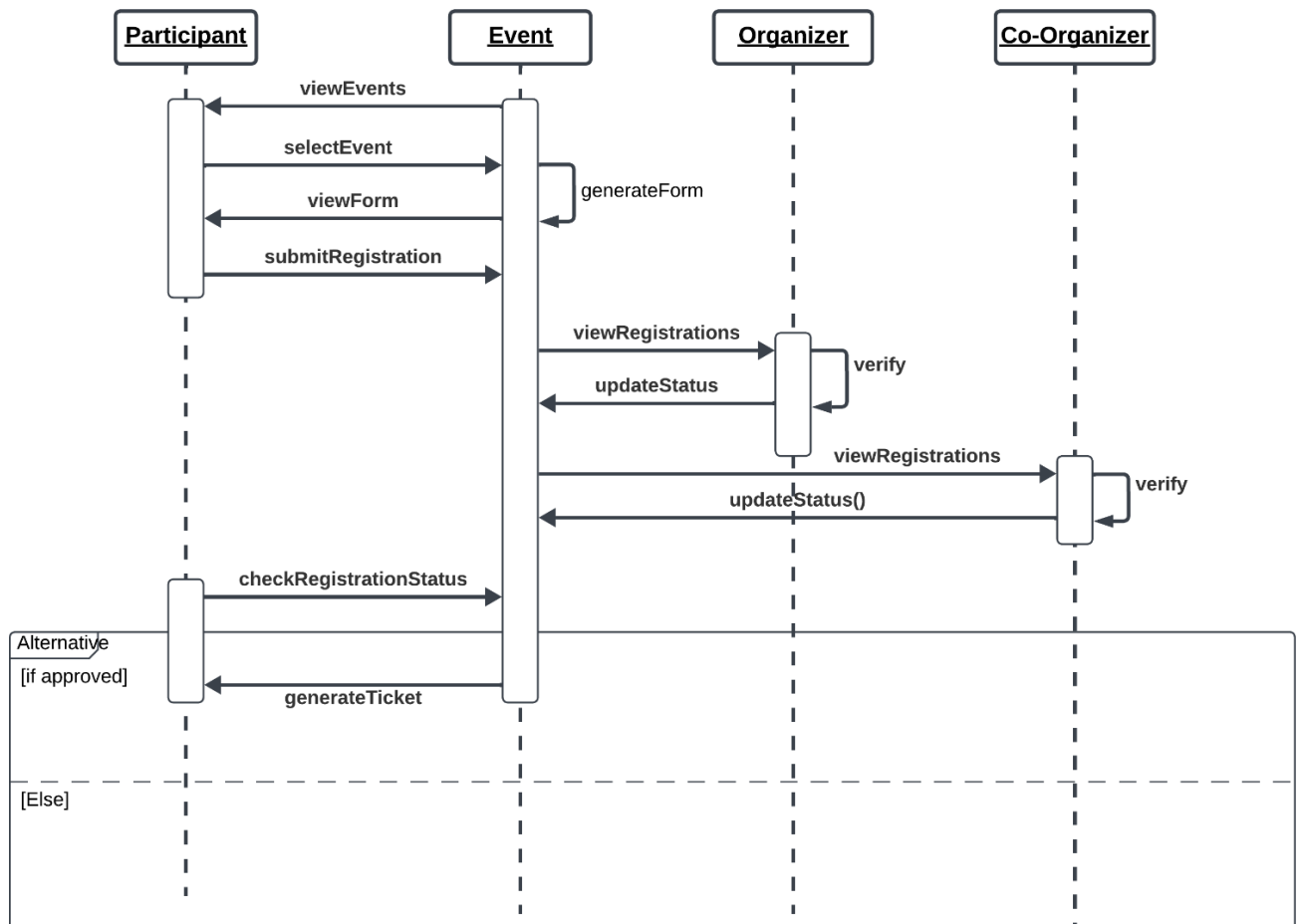


SEQUENCE DIAGRAM

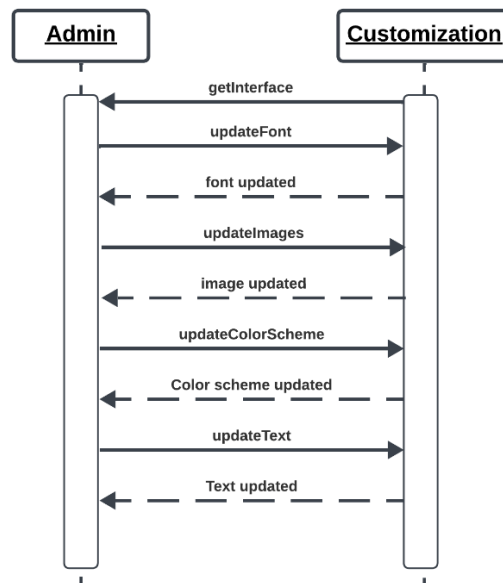
- Organizer Registration



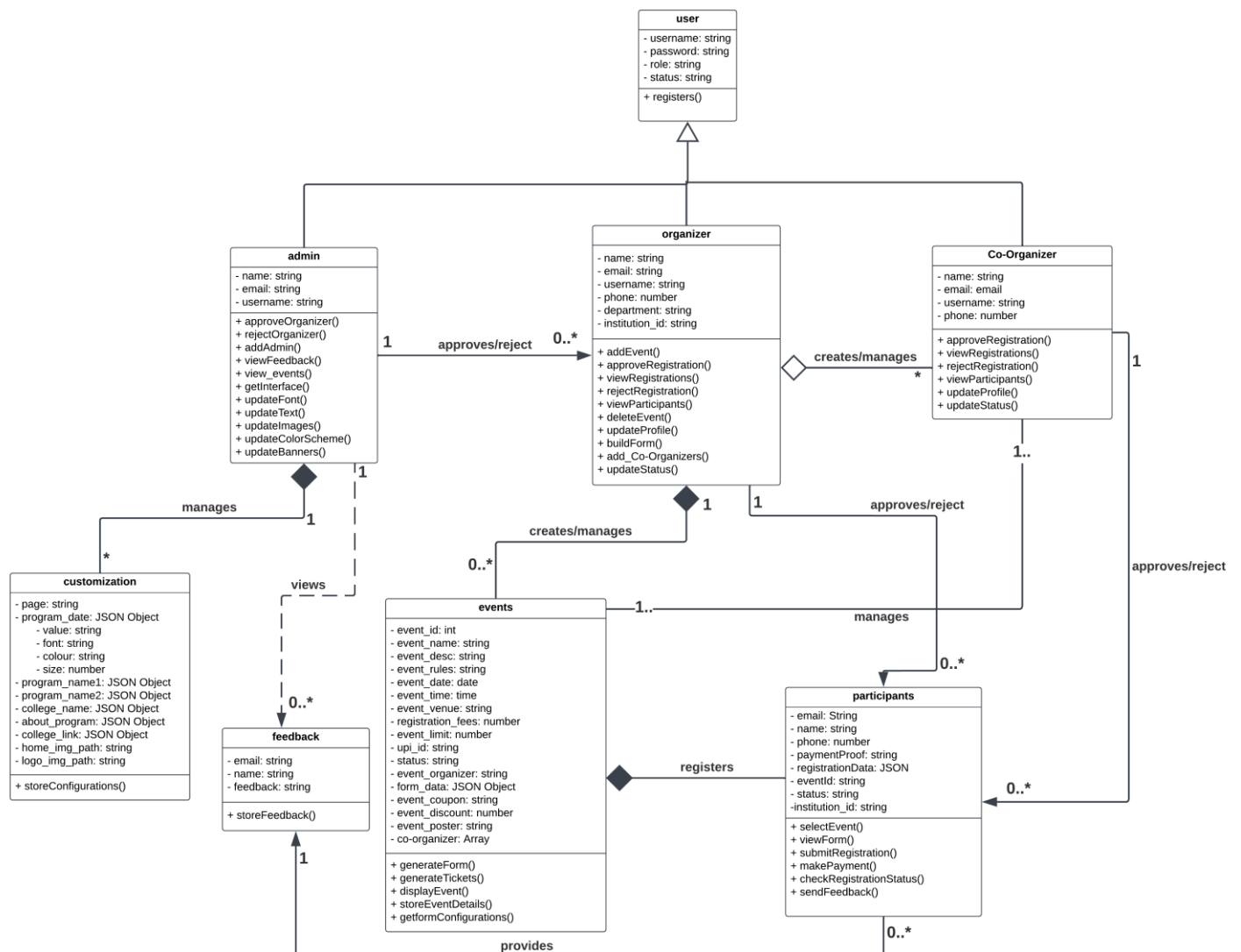
- Participant Event Registration



- Interface Customization

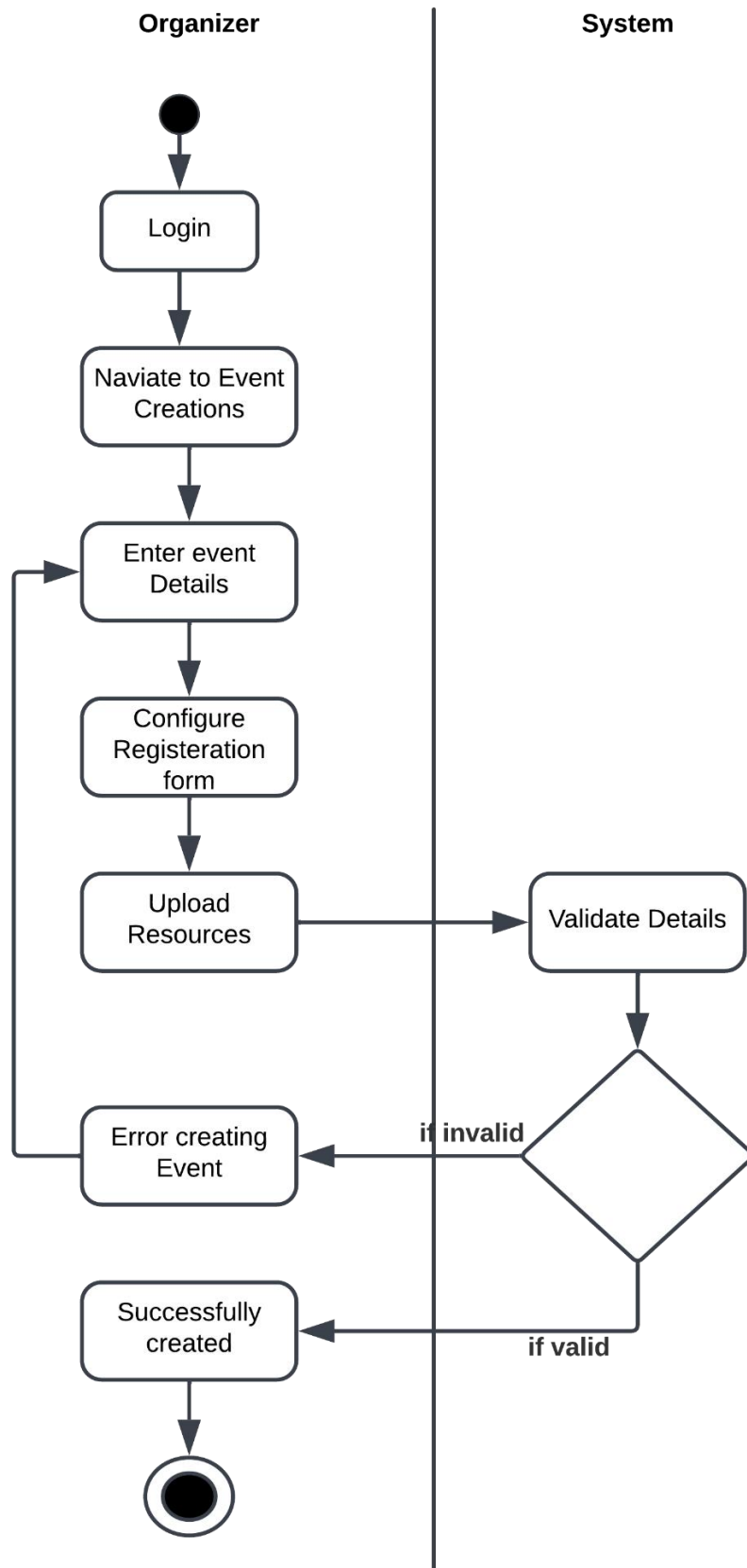


CLASS DIAGRAMs

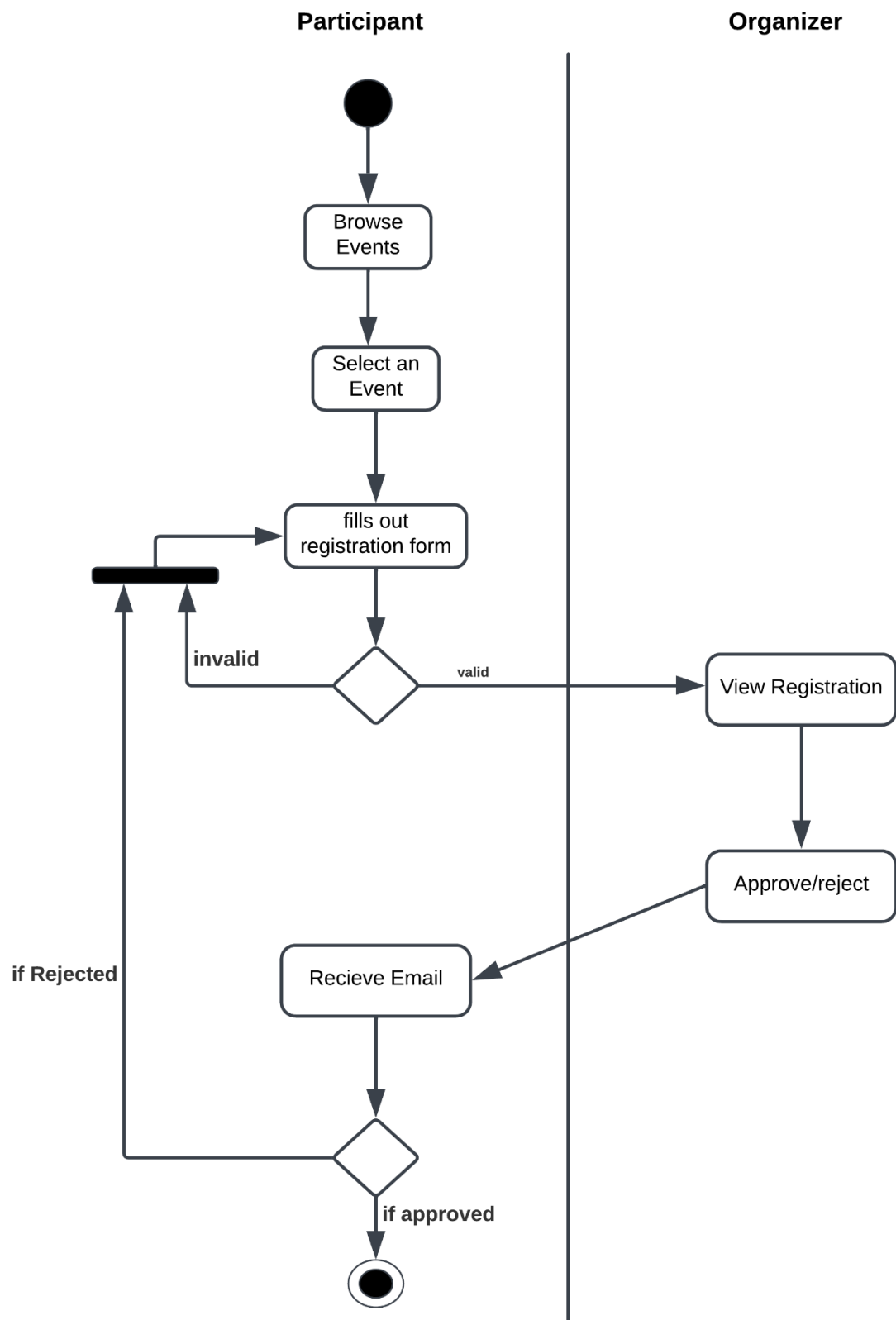


ACTIVITY DIAGRAM

- Event Creation



- Participant Event Registration



TECHNOLOGY/FRAMEWORK

Frontend:

We use HTML, CSS, JavaScript, and jQuery for building the frontend of the web application. HTML is used to create the structure of the web pages, providing ease of learning and integration with other web technologies. CSS is employed to style the pages, ensuring consistency in design and ease of maintenance. JavaScript adds interactivity to the pages, offering a smooth user experience with client-side interactions. jQuery, an essential library, is used for dynamic elements and simplifying complex tasks such as form validation and event handling. Additionally, the jQuery form builder tool is utilized for creating event-specific registration forms, which allows for easy customization to meet the specific needs of each event.

Backend:

The backend is powered by PHP, which handles the server-side logic of the application. PHP processes user requests, interacts with the database, and sends responses back to the client. Its flexibility, ease of use, and compatibility with various operating systems make it an ideal choice for web development. PHP is also an open-source language, reducing development costs. The backend logic includes functionalities like user authentication, event management, and data processing to ensure a smooth workflow for admins, organizers, and participants.

Database:

MongoDB is utilized as the database for the application. It is a NoSQL database, offering high scalability and flexibility for managing unstructured data. MongoDB supports fast data retrieval and provides robust data security, making it suitable for the dynamic nature of the event registration system. It stores various application data, including user profiles, event details, registration information, and payment history, ensuring that all necessary data is available for quick access and manipulation.

PROJECT FEASIBILITY

A project feasibility study is an essential analysis that determines whether a proposed project is viable in terms of technical, economic, and operational factors. It helps assess whether the project can be completed within the defined scope, timeline, budget, and available resources. A feasibility study also highlights potential challenges and ensures that the project can be executed successfully before substantial resources are allocated.

Technical Feasibility

The CampusConnect - Event Registration System is technically feasible due to the use of widely accessible and reliable technologies. The frontend utilizes HTML, CSS, JavaScript, and jQuery for creating an interactive and responsive user interface. The backend relies on PHP, which efficiently handles server-side processes such as user management, event creation, and registration handling. MongoDB is employed as the database, offering flexibility and scalability, making it ideal for managing diverse event data and user profiles. These technologies are well-supported, regularly updated, and widely adopted, ensuring the system will remain stable and secure. Additionally, the system's responsive design ensures compatibility across devices like computers, tablets, and smartphones, providing a seamless experience for administrators, organizers, and participants.

Economic Feasibility

The CampusConnect - Event Registration System is cost-effective as it leverages open-source technologies like HTML, CSS, JavaScript, PHP, and MongoDB, which do not incur licensing fees. These tools are widely used, enabling the easy availability of skilled developers and lowering labor costs. PHP's efficient processing reduces server load, leading to lower infrastructure costs, while MongoDB's NoSQL architecture allows for flexible data management without requiring expensive resources. The system also reduces operational costs by automating key tasks such as event registration, payment tracking, and user communications, thus minimizing manual work and human error. Hosting, maintenance, and support costs are the primary ongoing expenses, but these are outweighed by the long-term efficiency gains, reduced administrative workload, and improved user experience, making it a financially sound investment for educational institutions.

Operational Feasibility

The CampusConnect - Event Registration System streamlines event management and enhances operational efficiency by using modern web technologies. HTML, CSS, and JavaScript ensure a user-friendly and visually appealing interface, while jQuery simplifies dynamic elements like form building for custom event registration. On the backend, PHP effectively manages tasks such as user authentication, event scheduling, and registration processing. MongoDB provides flexible data storage, ensuring that the system can scale as event participation grows and user data becomes more complex. The system's modular architecture makes it easy to update and maintain, reducing downtime and keeping the platform aligned with evolving needs. By automating routine processes and ensuring accurate data handling, the system enhances overall productivity and reduces administrative overhead. To ensure long-term success, the system will require periodic updates, security patches, and skilled personnel for continuous maintenance, but its scalable design guarantees that it will continue to meet the dynamic needs of institutions.

SYSTEM ENVIROMENT

Hardware Environment

The hardware environment refers to the physical infrastructure necessary to support the development and deployment of the recommendation system.

Computing Resources:

Processor: 2 GHz, A powerful multi-core processor is recommended to handle the computational demands efficiently. Consider a processor with at least quad- core architecture for optimal performance.

Storage:

Sufficient Disk Space: 512 GB SSDMemory (RAM): 8 GB RAM: Sufficient RAM is crucial for efficiently handling large datasets andperforming machine learning tasks.

Internet Connectivity:

High-Speed Internet: A stable and high-speed internet connection is essential foraccessing online resources, libraries, and datasets during development.

Software Environment

1. Operating System

- Windows: Windows 10/11 or Windows Server 2019/2022.

2. Web Server

- Apache HTTP Server (Recommended): Version 2.4 or later.

3. Backend and Runtime Environment

- PHP: Version 8.0 or later.
 - Required PHP Extensions:
 - mongodb (for MongoDB integration)
 - json (for JSON handling)

4. Database

- MongoDB: Version 4.4 or later

5. Hosting Environment

- XAMPP/WAMP (Windows) or LAMP Stack (Linux).

6. Frontend Development

- HTML5: For structuring web pages.
- CSS3: For styling and layout.
- JavaScript: For client-side interactivity.
- jQuery: For dynamic form building and DOM manipulation.

7. Browser Compatibility

- Google Chrome: Version 90 or later.

8. Email Services

- SMTP Server:
- Gmail (via App Passwords) or custom SMTP for email notifications.

9. Development Tools

- Code Editor/IDE:
 - Visual Studio Code (with PHP, HTML, and JavaScript extensions).
- **Debugging Tools:**
 - Browser Developer Tools.
 - Xdebug for PHP.

DATABASE DESIGN

1. Users

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
username	String	Unique	Username of the user
password	String		Password to login
role	String		Role of the user
status	String		Status of the user(approved/reject)

2. Admin

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
username	String	Ref (Users: username)	Username of the admin
email	String	Unique	Email of admin

3. Organizers

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
username	String	Ref (Users: username)	Username of Organizer
email	String	Unique	Email of Organizer
phone	Number		Phone number of Organizer
department	String		Department of Organizer
institution_id	String		Institution Id image path

4. Events

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
event_name	String		Name of the event
event_desc	String		Short Description about event
event_rules	String		Rules & Information about event
event_date	Date		Date of event
event_time	Time		Event starting time
event_venue	String		Venue of Event
Registration_fees	Number		Registration fees of the event
event_limit	Number		Max no of participants allowed
upi_id	String		Upi ID to receive event fees
status	String		Status of the event(live/hold)
event_organizer	String	Ref (Organizers: username)	Username of Organizer who created the event
form_data	JSON Object		Event Registration form fields
event_coupon	String		Promo Code of the event
event_discount	Number		Discount percentage of event for successful promo code validation
event_poster	String		Img path of event poster
co-organizers	Array	Ref (Co_organizers: username)	Collection of Co Organizer's username for the event

4.Participants

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
event_id	Object ID	Ref (Events: _id)	Event ID of participant registered event
email	String	Unique	Email of Organizer
phone	Number		Phone number of Organizer
department	String		Department of Organizer
institution_id	String		Institution Id image path
payment_proof	String		Payment Proof image path
status	String		Status of the Event Registration

5. Co-organizers

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
username	String	Ref (Users: _id)	Username of the Co-organizer
name	String		Name of the Co-Organizer
email	String		Email of Co-Organizer
Phone	Number		Phone number of Co-Organizer

6. Feedback

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
email	String	Ref (Participants: email)	Email Id of the participant
event_name	String	Ref (Events: event_name)	Event Name for which the feedback is submitted
feedback	String		Feedback
timestamp	Number	Date & Time	TimeStamp at which the feedback is submitted

7. Interface

Attributes	Datatype	Constraints	Description
_id	Object ID	Unique	Autogenerated ID
page	String	Unique	Name of the Customizable page
page_element1	Object		Style Attributes of Element 1 in the page
page_element2	Object		Style Attributes of Element 2 in the page ... upto 'n'
page_img_path	String		Path of an image in the page

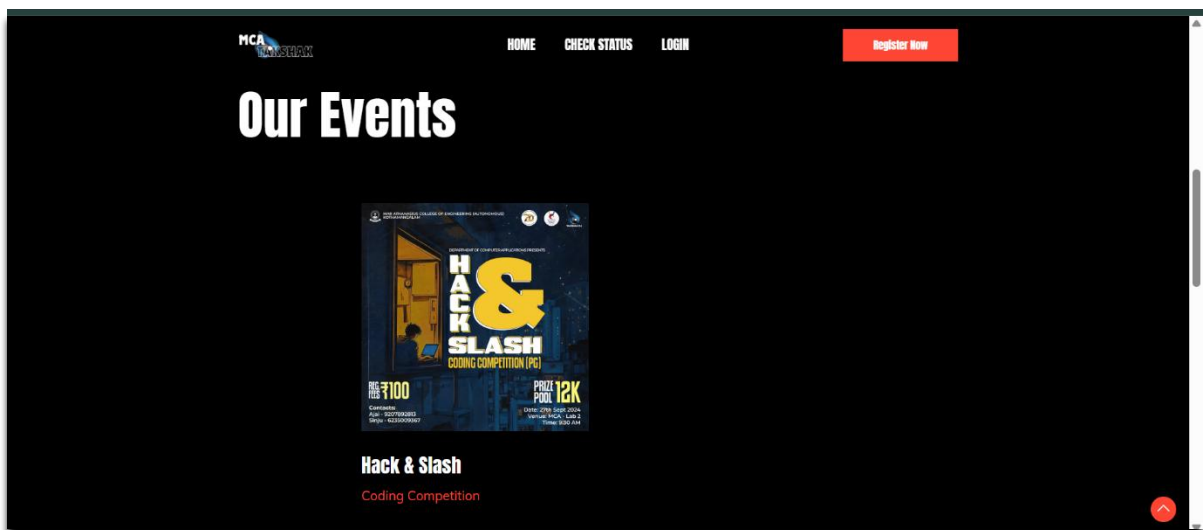
Elements are of Object datatype which consists of Style attributes such as Font Size, Font Family, Colour of the element and value of the element.

UI DESIGN

1. Home Page



2. Event Page



3. Payment QR Code

Department

Upload Institution ID

Choose File No file chosen


Promo Code

mace

Check

Promo code validated!

UPI Payment QR Code

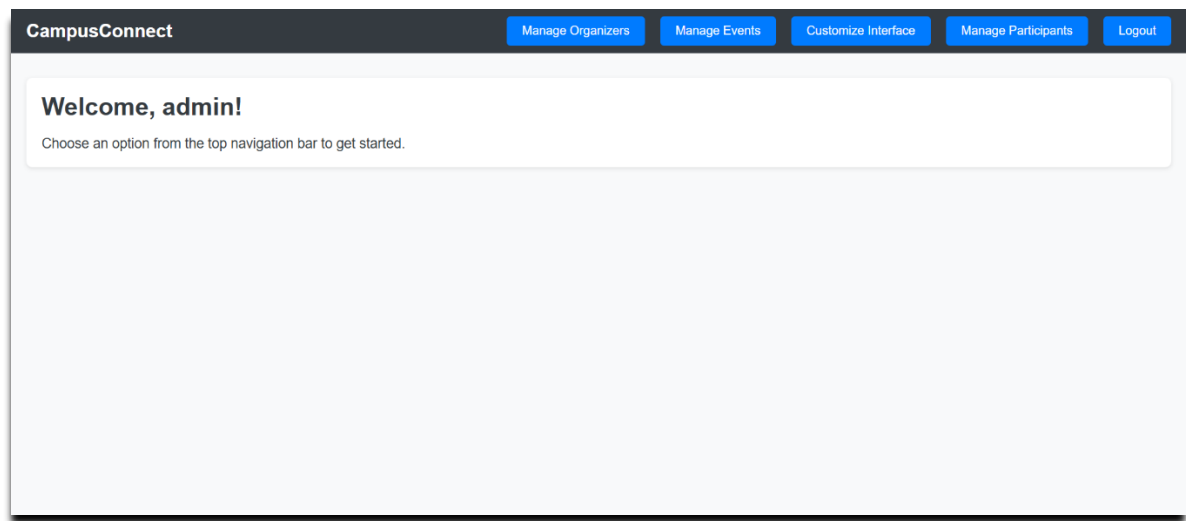


Upload Payment Proof

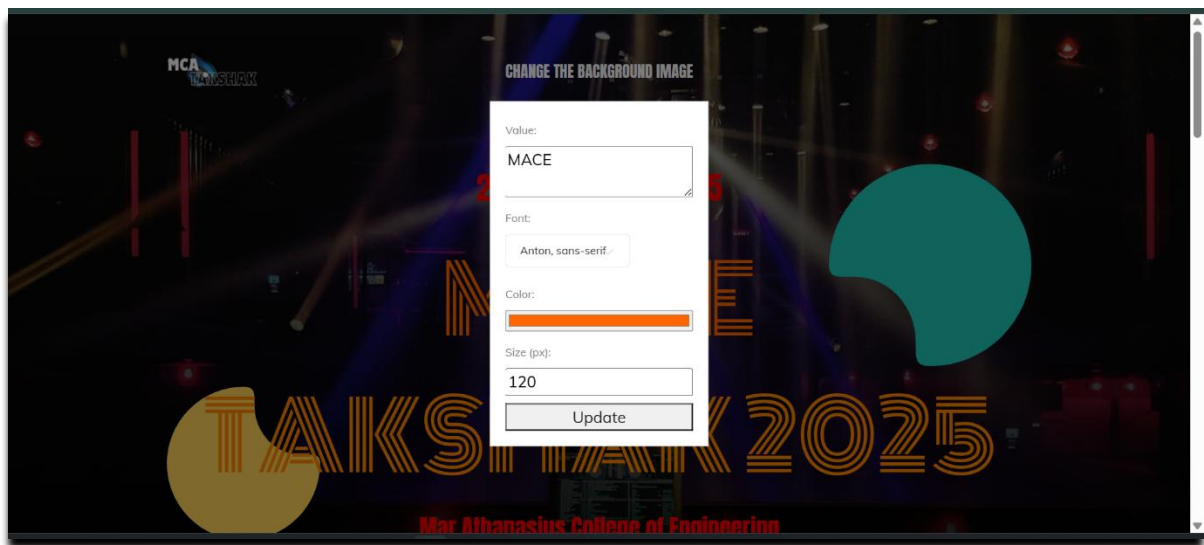
Choose File No file chosen

Submit

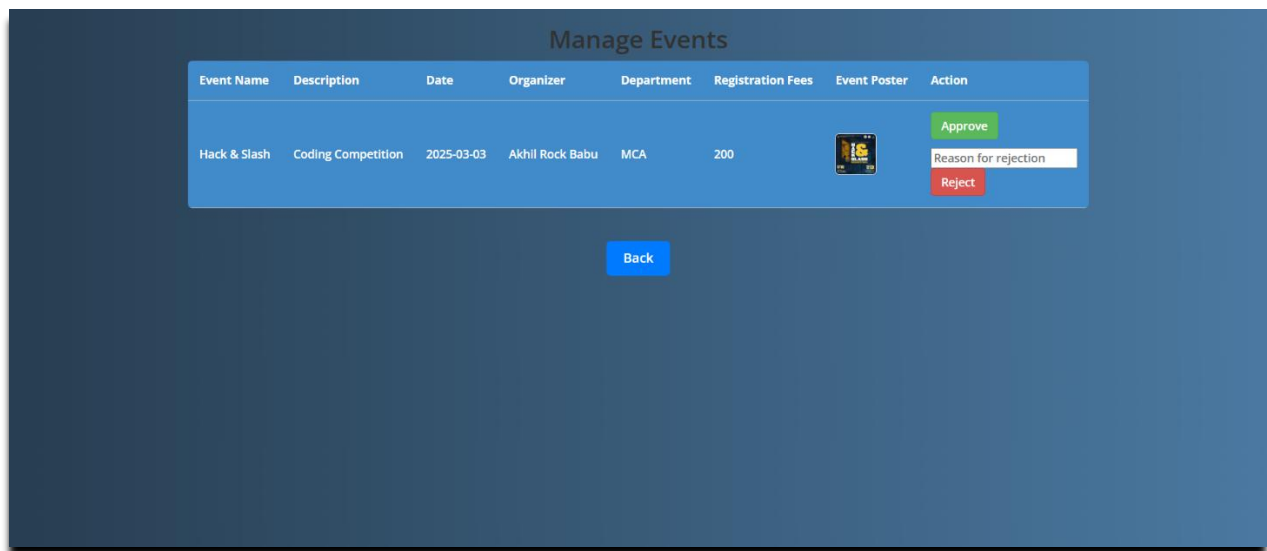
3. Admin Panel



4. Customization Page




5. Manage Events



6. Manage Organizers

Manage Organizers

Name	Email	Department	Institution ID	Action
Akhil Rock Babu	m23ca011@mace.ac.in	MCA		<div><div>Approve</div><div>Reason for rejection</div><div>Reject</div></div>

Back

7. Organizer Panel

CampusConnect

Create EventsApproved EventsUpdate ProfileLogout

Welcome, rockiey!

Organizer Panel

8. Form Builder

Design Your Event Form

Name

Name

Email

Email

Phone No

Phone No

Institution ID Proof

Institution ID Proof

Enter label

Text

×

Enter label

Text

×

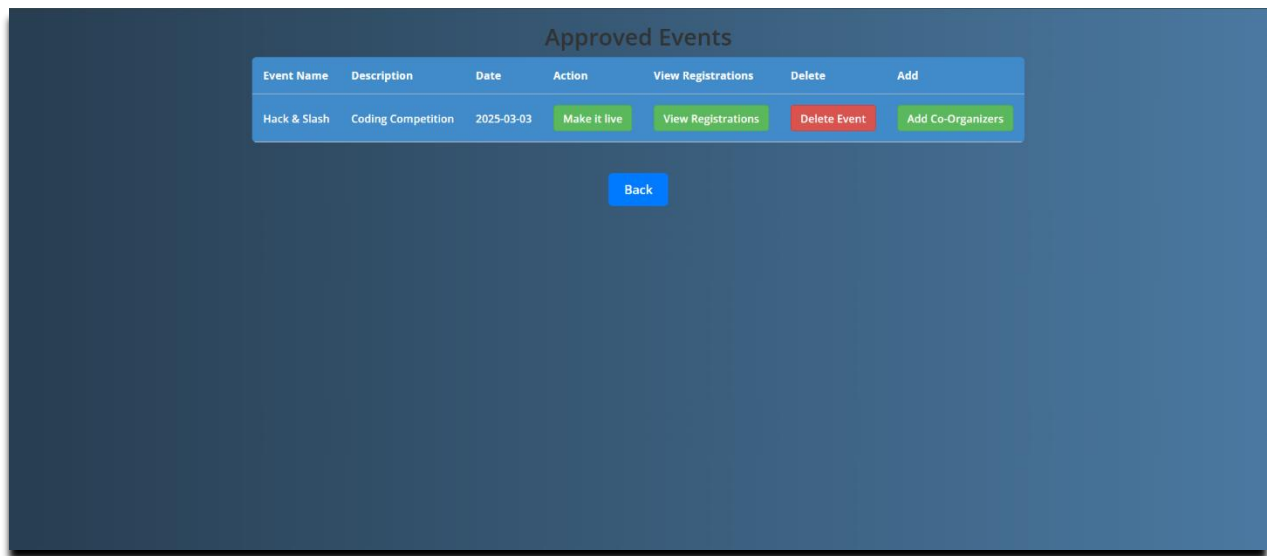
Add more fields to the form:

Add Field

Save Form



Back to Home

9. Approved Events Page



10. View Registration Data

Participant Registration Data

name	Akhil Rock Babu
email	akhilpes2002@gmail.com
phone	2345679560
Department	MCA
InstitutionID	
PaymentProof	
status	approved

Approve

Reason for Rejection: Reject

11. Add Co-Organizer Form

Add Co-Organizer

Add Co-Organizer

Back

12. Check Status Page

Check Status

Send OTP

Time left: 40 seconds

Check Status

Back to Home

13. View Status Page

View Status

Event Name	Event Description	Event Venue	Event Date	Event Time	Status	View Ticket
Hack & Slash	Coding Competition	MCA Lab 2	2025-03-03	07:04 PM	Approved	View Ticket

Back

14. Ticket Generation Page

MAR ATHANASIOUS
COLLEGE OF ENGINEERING
KOTHAMANGALAM, ERELA, 686 666
Ph : 0485 282340, 0485 282348

ADMIT ONE

AKHIL ROCK BABU

Admission No: M23CA011
Department : Computer Applications
Batch : CA 2023

Principal

Date of Issue: 04/08/2023
67c1920931a765bbb20ee59a

Monday March 3rd 2025

Hack & Slash
Akhil Rock Babu

07:04 PM

MCA Lab 2 © College Name

ADMIT ONE

ADMIT ONE

Download Ticket as PDF

LITERATURE REVIEW

Paper 1: J. R. V. Jeny, P. Sadhana, B. J. Kumar, S. L. Abhishek and T. Sai Chander, "A Web based-College Event Management System and Notification Sender," 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2022, pp. 1434-1438, doi: 10.1109/ICIRCA54612.2022.9985774.

Event management systems (EMS) have evolved from manual processes to automated web-based solutions, improving efficiency, user experience, and security. Various studies have explored different approaches to enhance EMS functionality. Some researchers have introduced barcode-based validation for secure participant verification, while others have implemented background authentication codes to improve security. Automated scheduling, notification alerts, and real-time data management have also been incorporated to reduce the burden on organizers.

The Model-View-Controller (MVC) architecture is widely used in EMS for its modular design, improving maintainability and scalability. SQL-based databases such as MySQL and SQLyog ensure efficient data handling, while technologies like Java Server Pages (JSP) and Servlets enable dynamic web interactions. Security remains a key focus, with researchers highlighting the importance of encryption techniques and authentication mechanisms. Performance optimization through real-time data processing methods has also been explored to enhance system responsiveness.

Future advancements in EMS may include AI-driven analytics for event forecasting, blockchain-based security for fraud prevention, and cloud integration for global accessibility. The continuous evolution of EMS is expected to further streamline event management, making it more efficient and user-friendly.

Paper: R. F. Olanrewaju, B. U. I. Khan, M. A. Morshidi, F. Anwar and M. L. B. M. Kiah, "A Frictionless and Secure User Authentication in Web-Based Premium Applications," in IEEE Access, vol. 9, pp. 129240-129255, 2021, doi: 10.1109/ACCESS.2021.3110310.

User authentication is a critical aspect of web-based applications, ensuring security while maintaining a seamless user experience. Traditional authentication mechanisms, such as username-password combinations, have proven vulnerable to attacks like phishing and brute force attempts. To address these challenges, researchers have explored advanced authentication solutions, integrating multi-factor authentication (MFA), biometric verification, and behavior-based authentication models.

Frictionless authentication has emerged as a promising approach to enhance security without disrupting the user experience. Studies highlight the effectiveness of behavior-based profiling, which leverages login patterns, device characteristics, and geolocation data to assess user authenticity. Risk-based authentication mechanisms dynamically adjust security measures based on contextual factors, reducing reliance on static credentials.

Several research efforts focus on multi-factor authentication systems, incorporating OTP verification, biometric data, and cryptographic tokens to strengthen access control. AI-driven authentication methods utilize machine learning algorithms to detect anomalies and unauthorized access attempts, enhancing system resilience against cyber threats.

Future advancements in authentication systems are expected to integrate blockchain-based identity management, federated authentication across platforms, and AI-driven fraud detection techniques. As web applications continue to evolve, secure and user-friendly authentication mechanisms will remain a key priority in ensuring data protection and accessibility.

Paper 3: K. T. Fathimath Hanan, T. Henna Sherin, U. Lulu Shadin, K. C. Dilshad and A. P. Sulthana Rinsy, "Streamlined Application for Managing College Events," 2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET), Nagpur, India, 2024, pp. 1-6, doi: 10.1109/ICICET59348.2024.10616323.

Event management in educational institutions has transitioned from traditional manual coordination to sophisticated web-based platforms that streamline planning, execution, and evaluation. Various research studies have explored how technological advancements have improved efficiency, security, and user experience in event management systems.

Several studies have focused on enhancing user accessibility and engagement by integrating modern web technologies. Many event management systems utilize Flutter for front-end development due to its cross-platform compatibility and user-friendly interface, while Python and Django are often employed for back-end functionalities to manage data flow and system logic. Real-time databases such as Firebase and MySQL enable seamless data storage, quick retrieval, and instant updates, ensuring efficient event registration, tracking, and notifications.

Security is a key aspect of event management systems. Researchers have explored different authentication techniques, including OTP-based login verification and encrypted data storage, to protect user information and prevent unauthorized access. These security measures enhance trust and reliability, ensuring data confidentiality and integrity.

Automation plays a crucial role in modern event management systems. Features such as automated scheduling, real-time notifications, and event evaluation modules have been developed to reduce manual effort and improve the overall event experience. Some studies have also introduced audience polling mechanisms, allowing participants to provide real-time feedback, thus enhancing engagement and decision-making.

Additionally, artificial intelligence (AI) and big data analytics are being leveraged to optimize event recommendations and attendance predictions. These technologies analyze past event data and user preferences to provide personalized event suggestions, helping organizers improve participation and resource allocation.

Future advancements in event management systems are expected to focus on integrating blockchain technology for credential verification, IoT-enabled real-time tracking for better logistics management, and AI-driven automation for smart decision-making. As educational institutions continue to embrace digital transformation, event management systems will further evolve to provide a more secure, efficient, and engaging platform for organizing college events.

LITERATURE SUMMARY

TITLE	TECHNOLOGIES USED	FEATURES	KEY CONTRIBUTIONS
Jeny et al. (2022). Web-based College Event Management System.	MVC Architecture, SQL-based databases.	Automated event approval/rejection, Admin dashboard, Notification system.	Enhances event organization by automating approval processes and structuring data storage for efficient retrieval.
Fathimath Hanan et al. (2024). Streamlined Application for Managing College Events.	Flutter (Front-end), Python-Django (Back-end), Firebase/MySQL (Database).	Real-time event tracking, Customizable registration forms, OTP-based authentication.	Improves participant experience with dynamic registration forms, instant updates, and enhanced security measures.
Olanrewaju et al. (2021). A Frictionless and Secure User Authentication in Web-Based Premium Applications.	AI-driven authentication, Multi-factor authentication (MFA), Blockchain-based identity management.	Behavior-based profiling, Risk-based authentication, Role-based access control.	Strengthens system security by incorporating multi-layered authentication mechanisms.

Web-based event registration and management systems have evolved with advancements in automation, security, and real-time data processing. Various studies have explored different methodologies to enhance efficiency, streamline user authentication, and improve event coordination. Several key features and functionalities in modern event registration systems are drawn from these studies, ensuring seamless event organization, secure user authentication, and enhanced user experience.

Jeny et al. (2022) present a web-based college event management system that automates event registration, scheduling, and notifications while incorporating authentication codes for security. The implementation of an automated event approval and rejection mechanism enables administrators to efficiently review and manage events. The study also highlights the use of the Model-View-Controller (MVC) architecture for a modular and scalable design, ensuring better system maintainability. Additionally, SQL-based databases are utilized for efficient data storage and retrieval, a principle that aligns with the structured storage and management of event-related information.

Fathimath Hanan et al. (2024) emphasize real-time updates, enhanced user engagement, and authentication in event management systems. The integration of real-time databases facilitates instant event tracking and notifications, ensuring up-to-date participant information. Dynamic event registration systems benefit from customizable form builders, allowing organizers to tailor event-specific registration forms based on requirements. Security measures such as OTP-based authentication enhance access control, providing a secure environment for users to verify their registration status and access event-related details.

Olanrewaju et al. (2021) focus on frictionless and secure authentication mechanisms, highlighting multi-factor authentication, AI-driven fraud detection, and behavior-based profiling. While AI-based authentication enhances security, essential security principles such as email OTP verification and role-based access control play a crucial role in preventing unauthorized access. These measures contribute to a secure event management system, ensuring that sensitive participant and event information is well protected.

The adoption of these research-backed features enables the development of an efficient and secure event registration and management system. The integration of automated approval processes, dynamic event creation, real-time updates, and secure authentication mechanisms enhances usability and operational efficiency, providing a streamlined experience for administrators, organizers, and participants.