-

FDBMS Mini Project Report

on

UniVent- Your University Event Partner

Submitted by

Batch A2 - Group 2

Project Members

Divya Yokar (1032232628)

Arnav Nikumbh (1032232650)

Rishi Raj Thakur (1032232721)

Naitri Panchal (1032232799)

Under the Guidance of

Dr. Sukhada Bhingarkar

Department of Computer Engineering and Technology
MIT World Peace University, Kothrud,
Pune 411 038, Maharashtra - India
2024-2025

Acknowledgement

We would like to express our sincere gratitude to all those who have supported and guided us throughout the development of our DBMS Mini Project — Club Management System.

First and foremost, we are deeply thankful to Dr. Sukhada Bhingarkar, Associate Dean of External Affairs and our subject teacher for Database Management Systems, for her constant guidance, encouragement, and valuable insights during the course of this project. Her expertise and timely feedback played a crucial role in shaping our understanding and execution of the concepts involved.

We also extend our thanks to Dr. Madhuri Bhalekar, Program Coordinator, CSBS, for her support throughout our academic journey.

We are grateful to Dr. Balaji Patil, Program Director, for providing us with a conducive learning environment and the resources required for successful project completion.

Our sincere appreciation goes to Dr. Sharmishta Desai, Associate Dean of Academics, and Dr. Mangesh Bedekar, Dean of the School of Computer Engineering and Technology, for fostering a strong academic culture that encourages innovation and hands-on learning.

Finally, we would like to thank all faculty members and peers who supported us directly or indirectly in completing this mini-project.

Divya Yokar

Arnav Nikumbh

Rishi Raj Thakur

Naitri Panchal

Abstract

The Club Management System is a web-based application designed to streamline and digitize the management of student clubs within a college environment. It facilitates efficient handling of core operations such as member registration, event scheduling, participation tracking, and record maintenance. Developed using HTML, CSS, JavaScript for the frontend, and Flask with MySQL for the backend, the system ensures smooth interaction with a robust relational database. Key features include user authentication, event creation and approval workflows, role-based access control, and real-time dashboards tailored to students, club heads, and administrators. The database design follows ER modeling, normalization, and structured query implementation to maintain data consistency and integrity. The system replaces manual processes with automated, scalable workflows that enhance accessibility and transparency. This mini-project not only showcases the practical application of DBMS concepts but also provides a centralized platform that simplifies club-related operations while fostering student engagement and collaborative event management.

List of Figures

Contents

Chapter 1
Introduction

In modern academic institutions, student clubs play an essential role in fostering holistic development by promoting leadership, collaboration, and co-curricular engagement. However, as clubs expand in both size and activity, managing their operations through manual processes becomes increasingly difficult. Challenges such as disorganized event planning, inefficient record-keeping, and lack of centralized communication often hinder productivity and transparency.

To address these issues, we developed **UniVent**—a comprehensive Club Management System designed specifically to cater to the needs of student clubs at the institutional level. UniVent aims to streamline various aspects of club administration, including member management, event scheduling, participation tracking, and approval workflows. By providing a unified digital platform for students, club coordinators, and administrators, the system enhances collaboration and accountability across all levels.

UniVent is built using web technologies such as HTML, CSS, and JavaScript for the frontend, and integrates a MySQL database with PHP on the backend to ensure efficient data handling and secure transactions. The project not only applies core database principles like ER modeling, normalization, and relational schema design but also demonstrates how these concepts can solve real-world challenges faced in everyday college life.

1.1 Problem Statement

Managing college clubs manually often leads to disorganized event planning, delayed approvals, inefficient communication, and loss of important data. Club heads typically rely on paperwork or informal channels such as messaging apps to register events, manage participants, and coordinate with faculty or administration. This results in a lack of centralized records, poor tracking of participation, and challenges in scheduling events or allocating venues and resources. Moreover, there is no unified platform for students to explore upcoming events or register easily. The absence of an integrated system makes it difficult to ensure transparency, accountability, and smooth operation across different clubs.

1.1.1 Need of the mini project

The need for a Club Management System arises from the increasing complexity and scale of managing college club operations. As student participation in extracurricular activities continues to grow, club coordinators face challenges in efficiently organizing events, maintaining accurate member records, and ensuring smooth communication. Traditional manual methods, such as maintaining spreadsheets, using messaging apps, or relying on paper-based documentation, are not only time-consuming but also error-prone and unscalable.

With multiple clubs operating simultaneously and conducting events throughout the academic year, the volume of data generated—ranging from student registrations and event schedules to venue and resource bookings—demands a more robust and centralized approach. A digital Club Management System addresses these issues by offering a structured and integrated solution that simplifies operations for students, club heads, and administrators alike.

Key benefits of the system include:

- **Centralized Data Management**: All essential information, including member profiles, event details, and participation records, is stored in a unified database that can be accessed securely by authorized users.

- **Improved Efficiency**: Routine tasks such as registering members, managing events, and tracking attendance are automated, saving time and reducing administrative overhead.

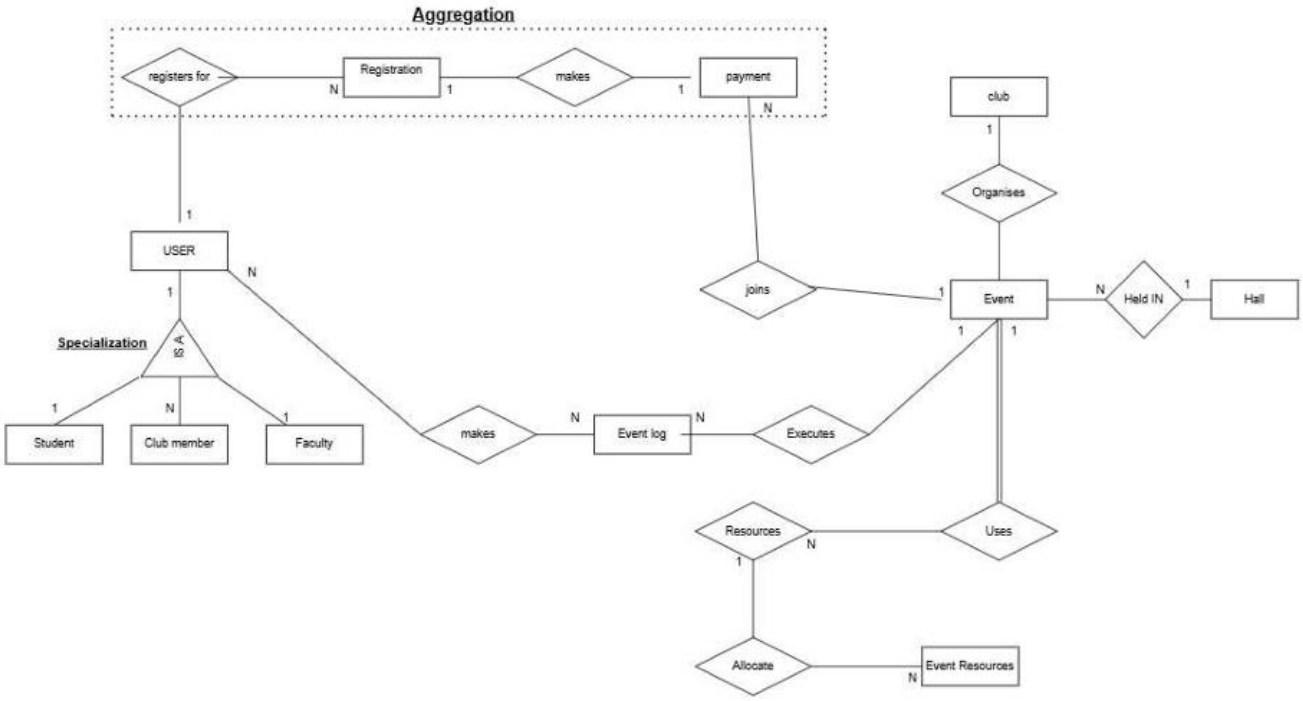
- **Enhanced Communication**: The platform facilitates better interaction between club organizers and members, with features such as notifications, event updates, and approval alerts.

- **Error Reduction**: By automating data entry and validation processes, the system minimizes the risk of human errors that can occur during manual record-keeping.

- **Role-Based Access Control**: Different user roles (students, club heads, admins) have customized access and functionality, ensuring a secure and organized workflow.

This mini project provides a practical opportunity to apply database concepts such as ER modeling, normalization, and relational schema design to a real-world scenario. It not only demonstrates technical proficiency but also promotes better management practices within student organizations. The system serves as a foundation for scalable and sustainable club operations, making it a valuable tool in any modern educational institution.

Chapter 2
ER Diagram



1. Users

This table stores details about all individuals who interact with the system, including students, faculty, mentors, coordinators, and club members. Each user is assigned a unique ID and a specific role which helps in determining their permissions and access rights in the system.

- **Key Fields:** user_id, name, email, password, role, department, join_date
- **Purpose:** User identification, login authentication, role-based access control.

2. Halls

The Halls table keeps a record of all the venues available on campus for conducting events. Each hall entry includes its name and capacity.

- **Key Fields:** hall_id, hall_name, capacity
- **Purpose:** Venue management and event scheduling.

3. Resources

Resources are physical items (like projectors, microphones, chairs) available in halls. This table links resources to halls and keeps track of the total and available quantities.

- **Key Fields:** resource_id, hall_id, resource_name, total_quantity, available_quantity
- **Purpose:** Inventory and resource allocation for events.

4. Clubs

This table holds essential details about student clubs such as the name, description, and leadership (president and mentor). It tracks whether the club is currently active.

- **Key Fields:** club_id, club_name, description, president_id, mentor_id, created_on, is_active
- **Purpose:** Managing student organizations and leadership structure.

5. Memberships

Memberships link users to clubs, defining the role they play within the club (e.g., member or president). It helps monitor who belongs to which club and since when.

- **Key Fields:** membership_id, user_id, club_id, role_in_club, joined_on
- **Purpose:** Tracking club membership history and roles.

6. Events

The Events table contains all planned club activities, including who scheduled them, where they'll take place, and their approval status.

- **Key Fields:** event_id, event_name, club_id, scheduled_by, hall_id, date, status, description
- **Purpose:** Event planning, status tracking, and logistical coordination.

7. Resource_Requests

This table manages requests made for resources required during events. It records who requested what, how many items were needed, and the current approval status.

- **Key Fields:** request_id, event_id, requested_by, resource_id, requested_quantity, status, allocated_quantity, request_date
- **Purpose:** Resource planning and allocation workflow.

8. Event_Registrations

Used to record which users have registered for specific events. This helps in managing participation lists and attendance.

- **Key Fields:** registration_id, event_id, user_id, registered_on
- **Purpose:** Tracking event attendees and engagement analytics.

9. Coordinators

This table stores information about users who act as department coordinators. These users are typically responsible for overseeing event and resource approval processes.

- **Key Fields:** coordinator_id, user_id, department
- **Purpose:** Authorizing activities and serving as points of contact for approval.

10. Approvals

Contains the outcome of approval requests (either for events or resources) reviewed by coordinators. It includes timestamps and optional comments for transparency.

- **Key Fields:** approval_id, coordinator_id, request_type, request_id, status, review_date, comments
- **Purpose:** Logging approval decisions and ensuring controlled workflows.

Chapter 3:Modules

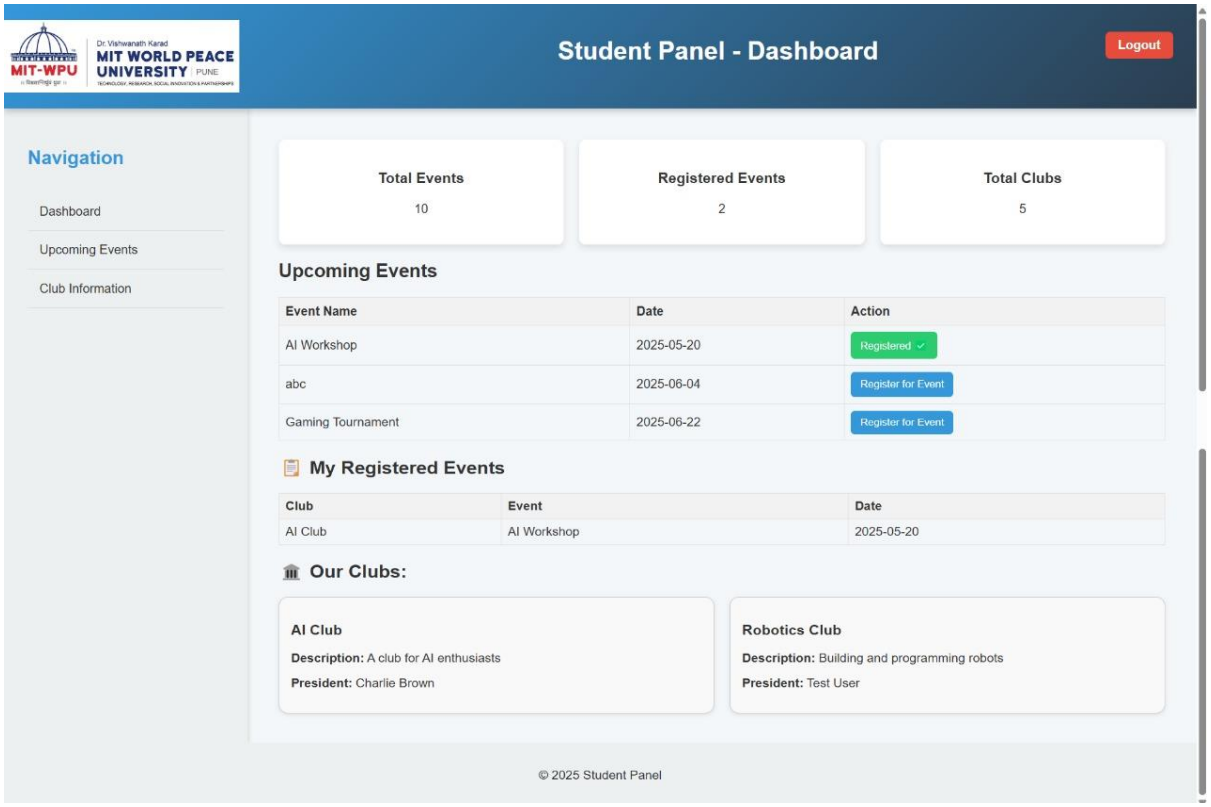
3.1 Login Page

The login page features the following components:

- **University Branding**: The page prominently displays the MIT-WPU logo and campus background, maintaining institutional identity.
- **User Authentication**: Users must enter their registered **email** and **password** to access the platform. This ensures secure and personalized access to features like event registration, club participation, and resource allocation.
- **"Remember Me" Option**: Allows the browser to save login credentials, improving usability for frequent users.
- **Responsive UI**: The layout is clean and user-friendly, promoting ease of access for all university stakeholders.

This interface is the **entry point** to the system and is connected to the **Users table** in the backend database. User credentials are validated before granting access, ensuring system security and user-specific functionalities

3.2 Student Dashboard



Key Features Displayed:

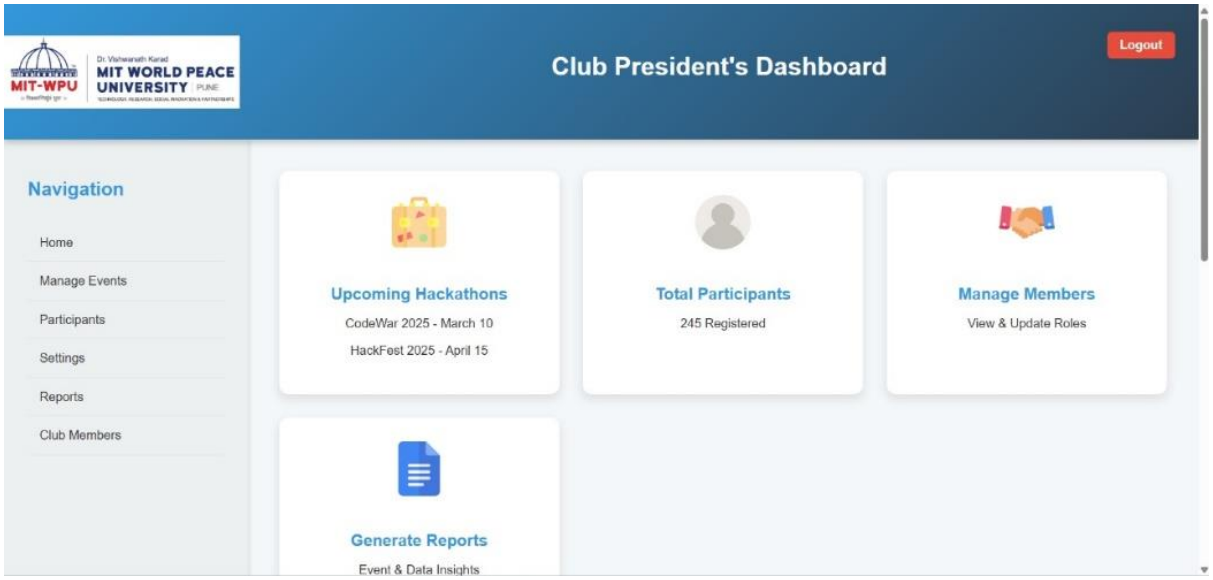
- **Dashboard Summary Cards**:
  - **Total Events**: Displays the number of events organized across all clubs.
  - **Registered Events**: Shows how many events the logged-in student has registered for.
  - **Total Clubs**: Indicates the total number of active student clubs in the system.

- **Upcoming Events Table**:
  - Lists all upcoming events along with their **names**, **dates**, and **registration status**.
  - Students can register for available events directly via the **"Register for Event"** button.
  - If a student is already registered, a **"Registered"** tag appears, avoiding duplicate registrations.

- **My Registered Events Section**:
  - Displays a list of events that the student has already enrolled in, along with the respective **club name** and **event date**.
  - Provides a personalized view for easy tracking of participation.

- **Sidebar Navigation**:
  - Simple navigation panel with links to **Dashboard**, **Upcoming Events**, and **Club Information** for quick access to relevant features

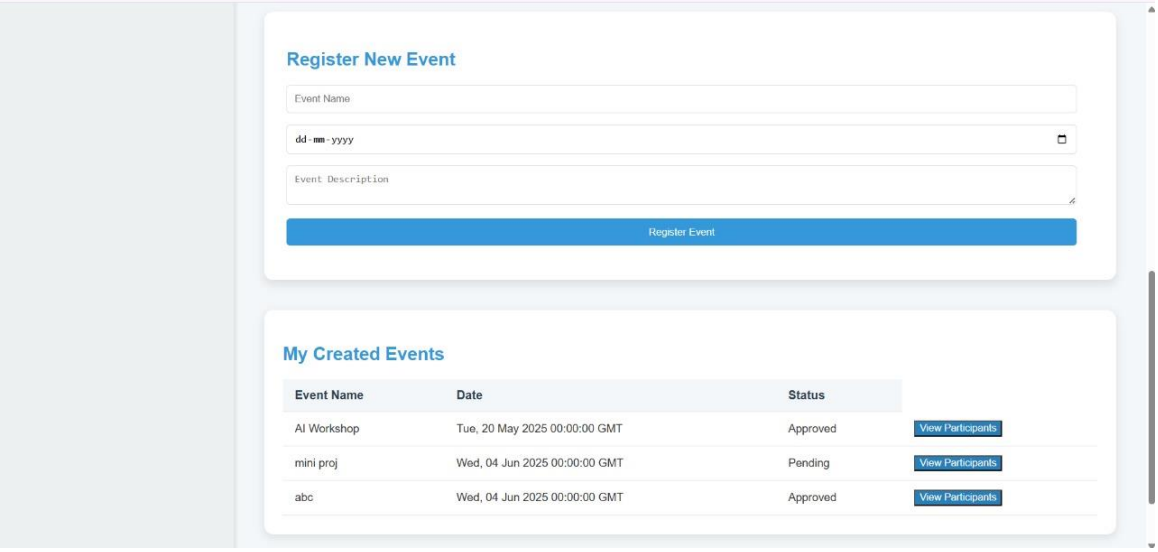
3.3.1 Club President Dashboard



- **Upcoming Hackathons:** Lists upcoming technical events such as CodeWar and HackFest with scheduled dates.
- **Total Participants:** Displays the total number of students registered for events under that club.
- **Manage Members:** Allows the club president to manage membership by viewing member lists and updating their roles (e.g., member, coordinator, vice-president).
- **Generate Reports:** Facilitates downloading or viewing detailed reports related to event participation, attendance, and insights, supporting administrative documentation and decision-making.

This dashboard helps club heads efficiently oversee event planning, team coordination, and performance tracking.

3.3.2 Event registration

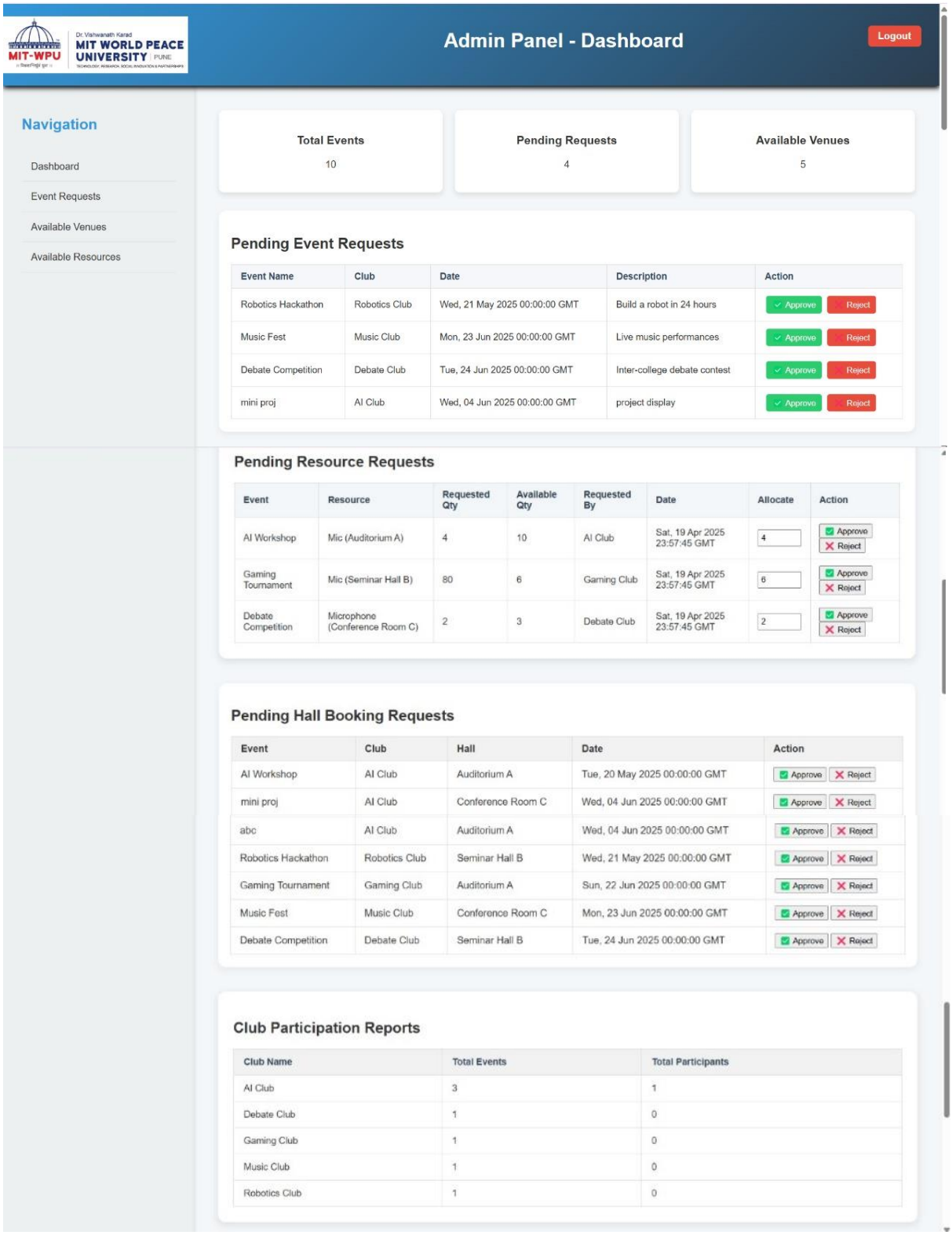


- **Register New Event Form**: Allows the user to input an event name, date (with a date picker), and a short description before clicking the **"Register Event"** button. This form directly connects to the backend DB to store event records.

- **My Created Events Section**: Displays a table listing all events created by the user:
  - **Event Name**: The title of the event (e.g., *AI Workshop*, *mini proj*).
  - **Date**: When the event is scheduled.
  - **Status**: Shows if the event is *Approved* or *Pending*.
  - **View Participants**: A clickable button to view a list of registered participants for each event.

This module supports event lifecycle management and provides visibility into each event's approval status and participation, helping club leaders track engagement and ensure smooth coordination.

3.4 Admin Dashboard



1. Admin Dashboard Overview

The first section of the dashboard provides a quick overview:

- **Total Events**: Indicates the total number of events currently listed in the system.
- **Pending Requests**: Shows how many event proposals are awaiting approval.
- **Available Venues**: Displays the number of venues currently free for allocation.

This panel helps administrators monitor the overall club activity and manage upcoming tasks efficiently.

2. Pending Event Requests
This section lists events that have been proposed by various clubs and are awaiting administrative approval. Each row displays:

- Event Name
- Club Name
- Date and Time
- Description
- Action Buttons (Approve/Reject)

This system ensures that event proposals go through a proper vetting process before being scheduled.

3. Pending Resource Requests
Clubs can request specific resources like microphones or AV equipment. This section logs:

- Requested Resources
- Quantity Requested vs. Quantity Available
- Requesting Club
- Allocation Field
- Action Buttons (Approve/Reject)

This feature prevents resource overallocation and ensures fair distribution based on availability.

4. Pending Hall Booking Requests
Before any event can be finalized, clubs must book appropriate venues. This module displays:

- Event Details
- Club and Hall Information
- Requested Date
- Action Buttons (Approve/Reject)

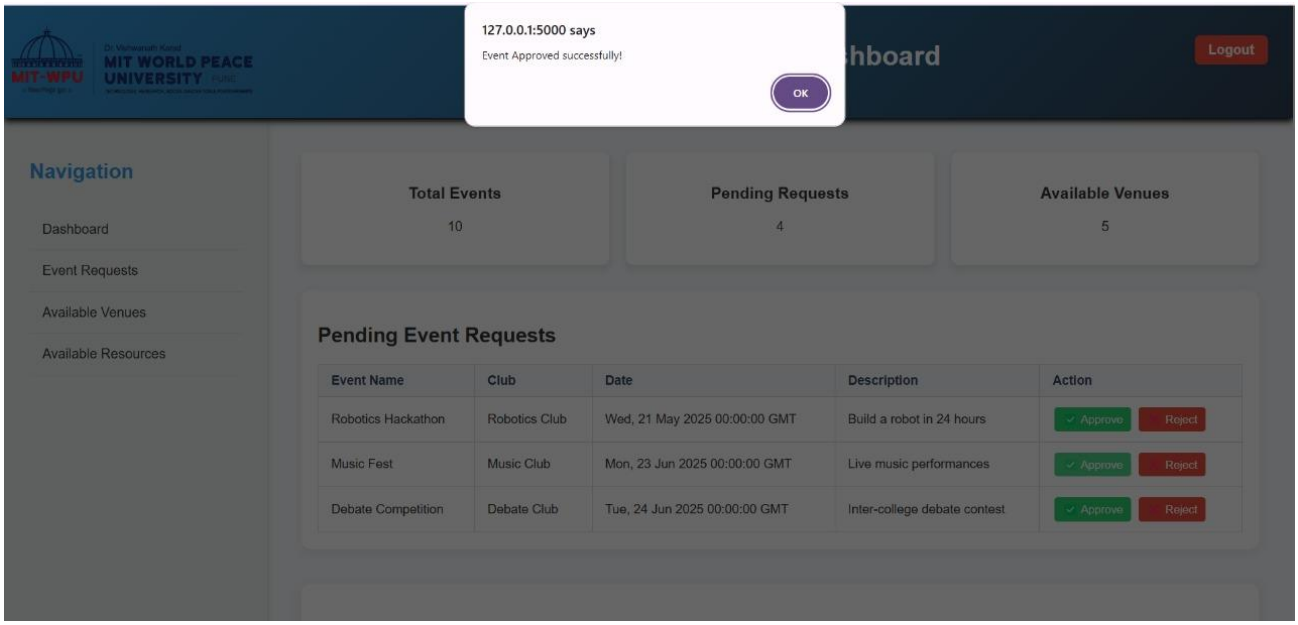
It provides centralized hall booking management to avoid scheduling conflicts.

5. Club Participation Reports
This report shows an overview of each club's involvement in organizing events:

- Total Events Conducted
- Total Participants

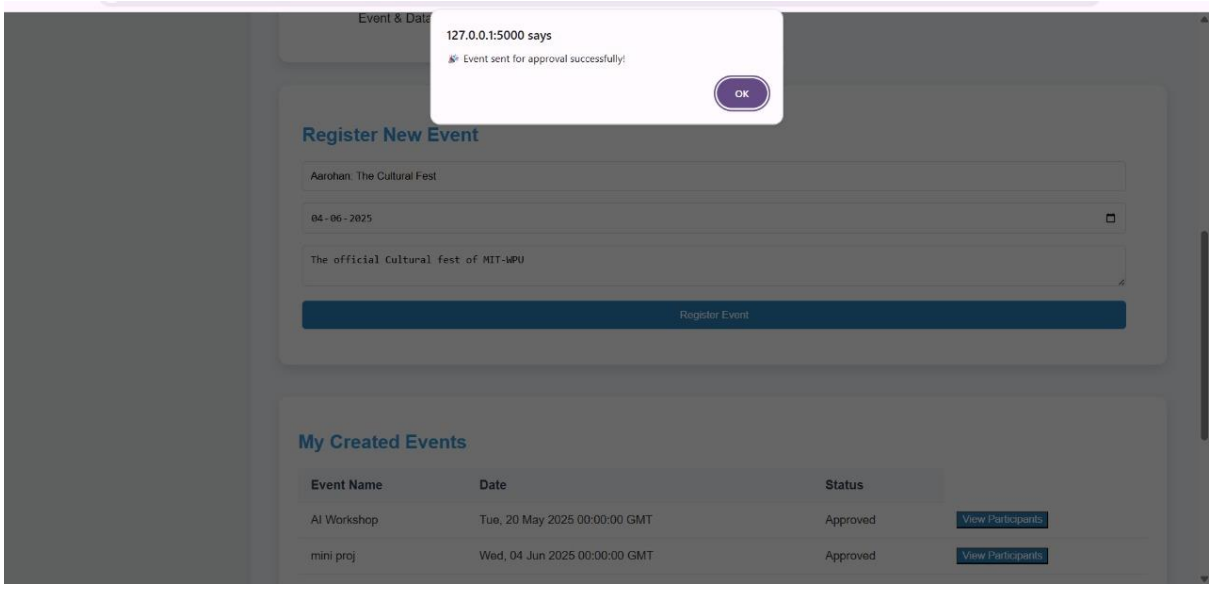
Chapter 4:Results

4.1 Event Approved



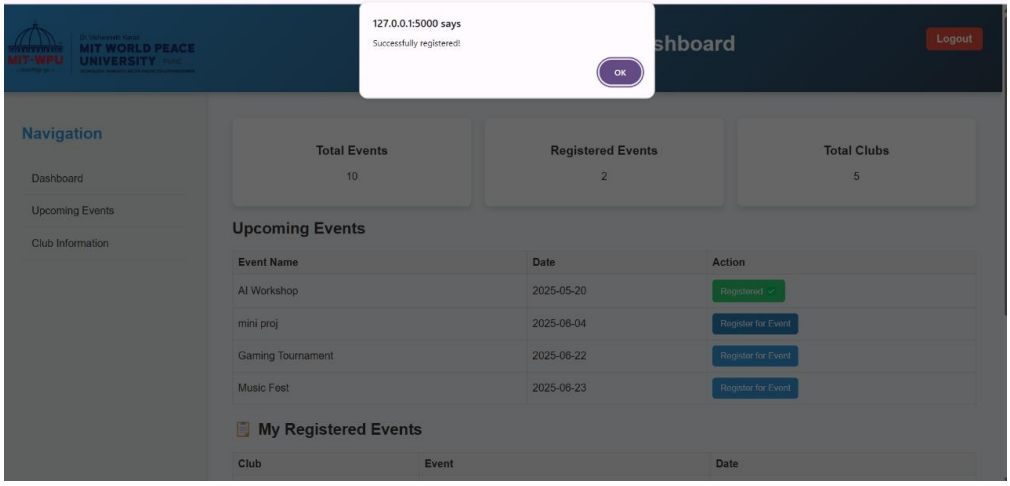
This result can be seen when admin approves the event after being requested president.

4.2 Request Approval



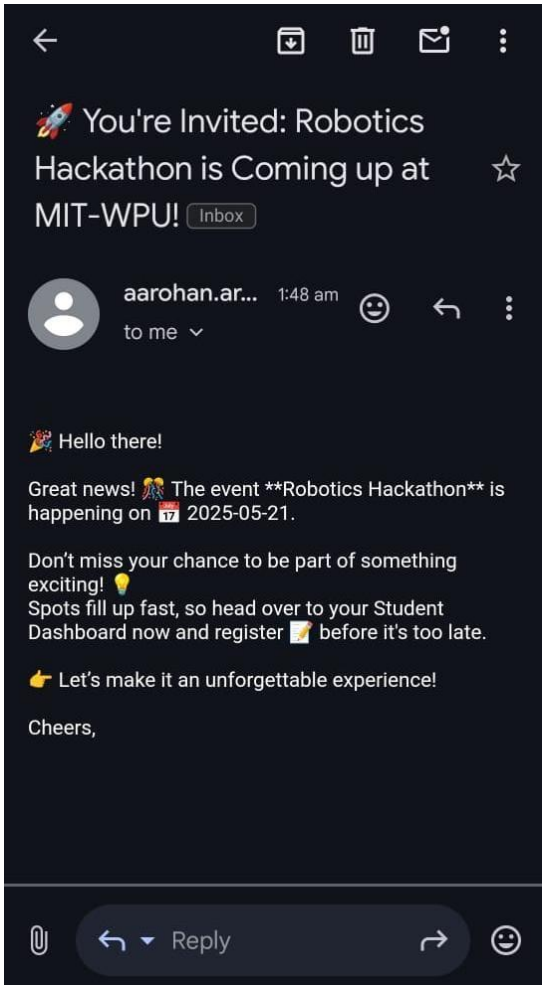
Here the club president request for event approval to the admin

4.3 Event Registration



When the student registered for the event this is the result

4.4 Notification sent via email



Email received by student after an event is approved

Future Prospects of the Project

As technology continues to evolve, the potential for enhancing UniVent with more intelligent and automated capabilities is vast. The system can be further expanded to offer a more seamless, data-driven experience for both users and administrators. Below are some promising directions for future development:

- **Automated Event Reports**: After an event concludes, the system can generate structured summaries using AI-based modules. These reports could include attendance metrics, event highlights, feedback insights, and other relevant data. This not only reduces the manual burden of report writing but also ensures consistency and timely record-keeping.
- **Streamlined Approval Workflow**: The existing multi-level event approval process can be improved by implementing secure digital authorization mechanisms. Faculty members can electronically validate proposals using their unique credentials. Once approved at one level, the system can automatically forward the request to the next authority, significantly improving response time and reducing bottlenecks.
- **Mobile App Integration**: Developing a mobile version of UniVent would allow students and organizers to access features on-the-go, increasing convenience and usability.
- **Analytics and Dashboards**: Incorporating visual dashboards to track event participation, member engagement, and resource utilization can help clubs make data-driven decisions.
- **Automated Notifications and Calendar Sync**: Integrating event reminders and calendar synchronization can help users stay informed and organized.
- **Feedback and Rating System**: Adding post-event feedback collection tools can enhance quality control and user satisfaction.
- **Budget and Resource Management**: Additional modules for tracking club finances and resource availability would support more professional and accountable event planning.

The proposed enhancements aim to make UniVent a more intelligent, autonomous, and user-friendly platform. By integrating automation, secure digital workflows, and advanced analytics, the system can evolve into a comprehensive solution that not only manages but empowers student clubs to operate at their full potential.

Conclusion

The UniVent Club Management System successfully addresses the administrative challenges faced by student clubs by providing a centralized, digital platform for managing club activities. Through its user-friendly interface and structured database integration, the system ensures efficient registration, event coordination, and participation tracking. It enhances communication between students, club heads, and administrators, making the entire process more transparent and organized. By applying core DBMS principles such as ER modeling, normalization, and SQL queries, the project demonstrates the practical relevance of database systems in solving real-world problems. UniVent serves as a scalable solution that simplifies routine tasks while promoting active student engagement within the campus community.

References

Student Project Guide. (n.d.). Event Management System Database Design. Retrieved from https://studentprojectguide.com/project-report/database-design/event-management-system-database-design

GeeksforGeeks. (2022). How to Design a Database for Event Management. Retrieved from https://www.geeksforgeeks.org/how-to-design-a-database-for-event-management

ResearchGate. (2024). Event Management System Project Report. Retrieved from https://www.researchgate.net/publication/380897786_EVENT_MANAGEMENT_SYSTEM_PROJECT_REPORT

Scribd. (n.d.). Event Management System Project Report. Retrieved from https://www.scribd.com/document/318921856/EVENT-MANAGEMENT-SYSTEM-PROJECT-REPORT-doc

Deb, A. (2024). Event Management Database Design Part 1. Medium. Retrieved from https://medium.com/@arpita_deb/event-management-database-design-part-1-5239620410c1

Tatibaev, M. (2023). Building an Event Management System: Designing the Blueprint, Crafting the Schema, and Executing. Medium. Retrieved from https://medium.com/@tatibaevmurod/building-an-event-management-system-designing-the-blueprint-crafting-the-schema-and-executing-43ad2e45568e

Drkušić, E. (2016). An Event Management Data Model. Vertabelo Database Modeler. Retrieved from https://vertabelo.com/blog/how-to-plan-and-run-events-an-event-management-data-model/

Governors State University. (2023). Event Management System. Retrieved from https://opus.govst.edu/cgi/viewcontent.cgi?article=1629&context=capstones

Shah, J. (2024). Event Management System. GitHub. Retrieved from https://github.com/jeel-shah24/Event-management-system