

TITLE OF PROJECT

UCS 503 Software Engineering Project Report

END-Semester Evaluation

Submitted by:

(102203594) SHAURYA JAIN

(102203597) DHRUV SINGH PUNDIR

(102203695) GURANSH SINGH

(102203747) ARSHDEEP ASWAL

BE Third Year, COE

Group No:1

**Submitted to
Dr. Nidhi chakravarthy**



**THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)**

Computer Science and Engineering Department

TIET, Patiala

November 2024

TABLE OF CONTENTS

S.No.	Assignment	Page No.
1. Project Selection Phase		
1.1	Software Bid	
1.2	Project Overview	
2. Analysis Phase		
2.1	Use Cases	
2.1.1	Use-Case Diagrams	
2.1.2	Use Case Templates	
2.2	Activity Diagram and Swimlane Diagrams	
2.3	Data Flow Diagrams (DFDs)	
2.3.1	DFD Level 0	
2.3.2	DFD Level 1	
2.3.2	DFD Level 2	
2.4	Software Requirement Specification in IEEE Format	
2.5	User Stories and Story Cards	
3. Design Phase (At least two significant cases of each diagram)		
3.1	Class Diagram	
3.2	Sequence Diagram	
3.3	Collaboration Diagram	
3.4	State Chart Diagrams	

4. Implementation

- 4.1 Component Diagrams
- 4.2 Deployment Diagrams
- 4.3 Screenshots

5. Testing

- 5.1 Test Plan
- 5.2 Test Cases
- 5.3 Test Reports

1. Project Selection Phase

1.1 Software Bid/ Project Teams

UCS 503- Software Engineering Lab

Group : 1

Dated:23rd August 2024

Team Name:

Team ID (will be assigned by Instructor):

Please enter the names of your Preferred Team Members. : Shaurya Jain, Dhruv Singh Pundir, Guransh Singh, Arshdeep Aswal

You are required to form **a three to four person** teams

Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used	Signature
Shaurya Jain	102203594			
Dhruv Singh Pundir	102203597			
Guransh Singh	102203695			
Arshdeep Aswal	102203747			

Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1.JavaScript

2.Python

3.C++

Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference: [*Write at-least one paragraph for each choice (motivation, reason for choice, feasibility analysis, etc.)*]

	Project Name	Unique Selling Point
First Choice	Virtual Event Management System (VEMS)	The Virtual Event Management System (VEMS) is a web-based platform that allows users to organize, manage, and attend virtual events like conferences, workshops, webinars, and social gatherings. The platform provides features like event creation, registration. It also includes analytics tools for organizers to track attendance, engagement, and feedback.
Second Choice	sports equipment management system	A Sports Equipment Management System is designed to manage and track the inventory of sports equipment in organizations such as schools, gyms, sports clubs, and rental agencies. This system helps in the efficient management of sports gear, ensuring that items are available, maintained, and tracked throughout their lifecycle. Here's a detailed description of the system:

Third Choice	Travel Buddy	<p>Website which helps college individual to find buddies who want to travel together to the same location for their journey which can be their home place so that they can have a companion with them and security of travelling along with reduced expenses.</p> <p>This mainly focuses on new individuals who do not know or have acquaintance who belong to the same hometown.</p>
Fourth Choice	Thapar Discussion Forum	<p>The Thapar Discussion Forum is a digital platform designed to facilitate communication and collaboration among students, faculty, and staff at Thapar Institute of Engineering and Technology (TIET). This forum serves as a centralized space for academic discussions, sharing resources, seeking help, and building a community within the institution.</p>

Additional Remarks/ Inputs

Please tell us about any other factors that we should take into consideration (e.g., if you really would like to work on a project for some particularly convincing reason).

1.2 Project Overview

Introduction:

The Event Management System (EMS) is a comprehensive platform designed for the Thapar Institute of Engineering and Technology (TIET) to streamline the management, registration, and reporting of events. The system aims to improve the organization of various events such as workshops, seminars, conferences, and cultural activities. With an intuitive user interface and seamless integration with existing institutional infrastructure, the EMS empowers event organizers, participants, and administrators to manage events more effectively and efficiently.

The system serves as a centralized hub for all event-related activities, allowing for smooth event creation, participant registration, scheduling, and reporting. By automating key processes, EMS reduces administrative burdens, improves user engagement, and ensures a seamless experience for all stakeholders.

Solution:

EMS provides a centralized, web-based solution for event management that incorporates essential features like event creation, registration, scheduling, notifications, and reporting. The system ensures that event organizers can manage and track events effortlessly while participants can register and receive notifications about upcoming events.

The backend is built with a focus on scalability and security, using a robust database (e.g., PostgreSQL) and server technologies. Integration with TIET's existing authentication systems ensures a secure and smooth user experience for students, faculty, and staff.

USPs (Unique Selling Points):

- Centralized Event Management: Streamlines the process of creating, managing, and tracking events with an easy-to-use interface.
- Real-time Notifications: Users receive timely notifications about event schedules, updates, and reminders through email/SMS.
- Role-Based Access Control: Ensures that only authorized users (event organizers, administrators) can perform specific actions within the system.
- Reporting and Analytics: Provides event organizers and administrators with powerful reporting tools to track event participation, feedback, and overall performance.

Objectives:

- User-Friendly Portal: A responsive and intuitive portal for users to easily register for events, manage their profiles, and receive notifications.
- Backend Performance: A high-performance backend built using technologies like Go and PostgreSQL to ensure scalability and reliability. The system supports multiple event categories and large user volumes..
- Event Scheduling: Accurate event scheduling and notifications with integration of calendar systems for easy tracking of event dates and times.

Look and Feel:

The EMS will feature a clean, modern design that prioritizes usability and accessibility. The website will be fully responsive, ensuring a smooth experience across devices such as desktops, tablets, and smartphones. The design will incorporate colors and styles that align with TIET's branding, creating a professional yet welcoming environment for users. Key actions, like event registration and user profile management, will be easily accessible and visually highlighted to guide users through their tasks without confusion. The design will focus on simplicity, with an emphasis on making interactions intuitive and efficient for both organizers and participants.

Scope:

- Event Creation and Management: The Event Management System (EMS) empowers event organizers to create and manage events at Thapar Institute of Engineering and Technology (TIET). This includes specifying event details such as title, date, time, venue, and description, and ensuring that the events are listed on the platform for visibility.
- User Registration and Profile Management: EMS allows users (participants, organizers, and administrators) to register and manage their profiles within the system. Users can create and update their personal information, making it easy to track event participation and manage registrations.
- Event Registration and Participation: The system facilitates event registration by allowing participants to sign up for various events. It ensures real-time updates on availability, providing confirmation upon successful registration, and allowing users to manage their event participation status.
- Event Scheduling and Notifications: EMS enables event organizers to set up schedules for upcoming events and send notifications to participants. Automated email and SMS notifications ensure users are informed about event reminders, schedule changes, and cancellations, ensuring everyone remains updated.
- Reporting and Analytics: EMS includes powerful reporting and analytics features that allow organizers and administrators to generate and view detailed reports. These reports provide insights into event participation rates, feedback summaries, and other relevant metrics to evaluate the success of events.
- Administrator Controls and Oversight: The system empowers administrators with the ability to monitor and manage event creation, user registrations, and overall system

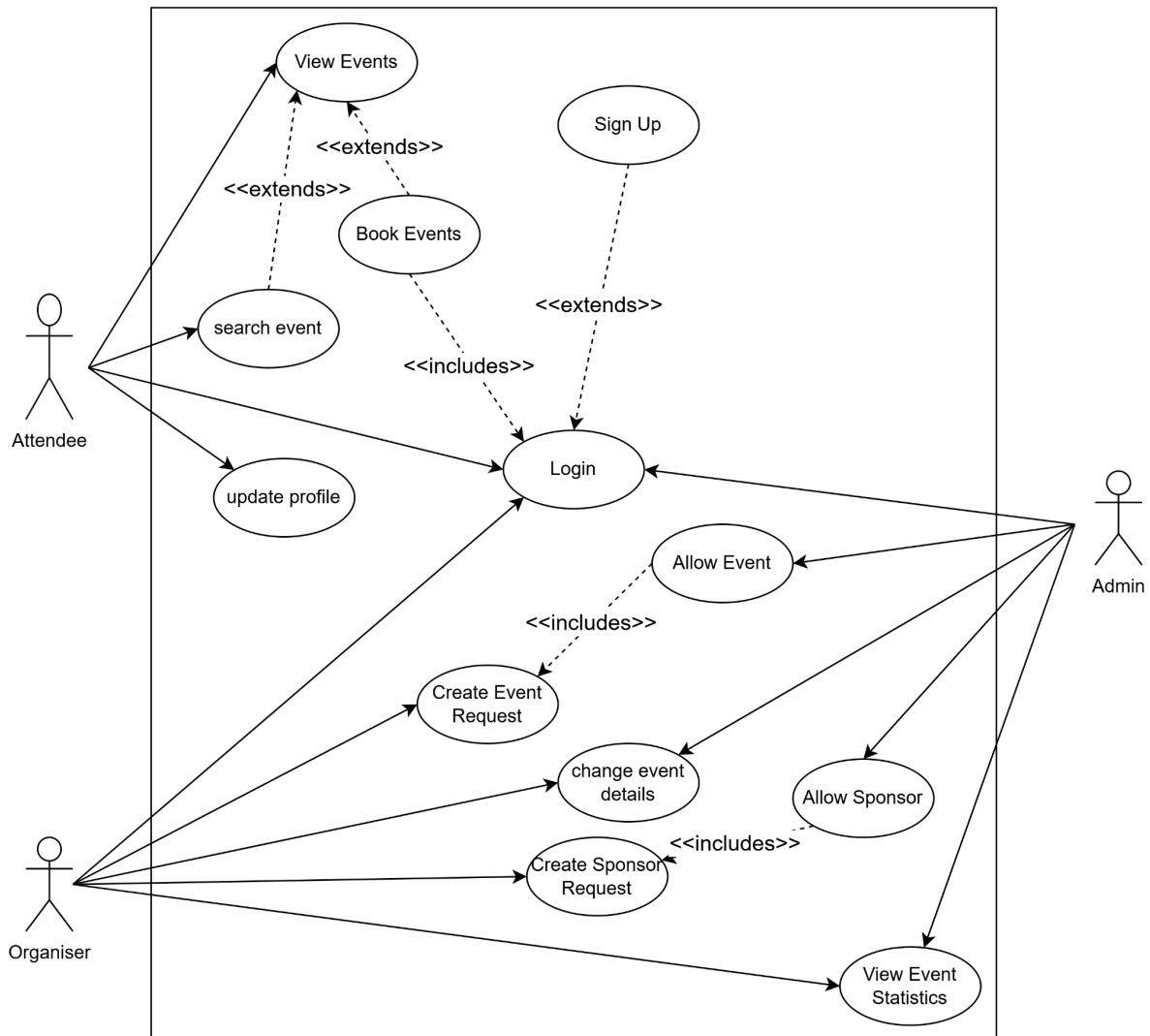
health. They can approve, reject, or modify event details and ensure smooth operations within the platform.

- Integration with TIET Authentication Systems: The EMS will only allow email provided by the institute to be used for registration to ensure secure systems.
- Scalability and Reliability: The EMS is designed to handle large-scale usage, supporting a growing number of users and events. The backend infrastructure is scalable and optimized for performance, ensuring that the system can handle peak loads during major events.
- Resource Allocation and Prioritization: By providing administrators and organizers with detailed event data and reports, the EMS helps optimize the allocation of resources for event planning and management. This functionality allows for more efficient management of event-related tasks, ensuring that time and effort are focused on the most impactful activities.

2. Analysis Phase

2.1 Use Cases

2.1.1 Use Case Diagram



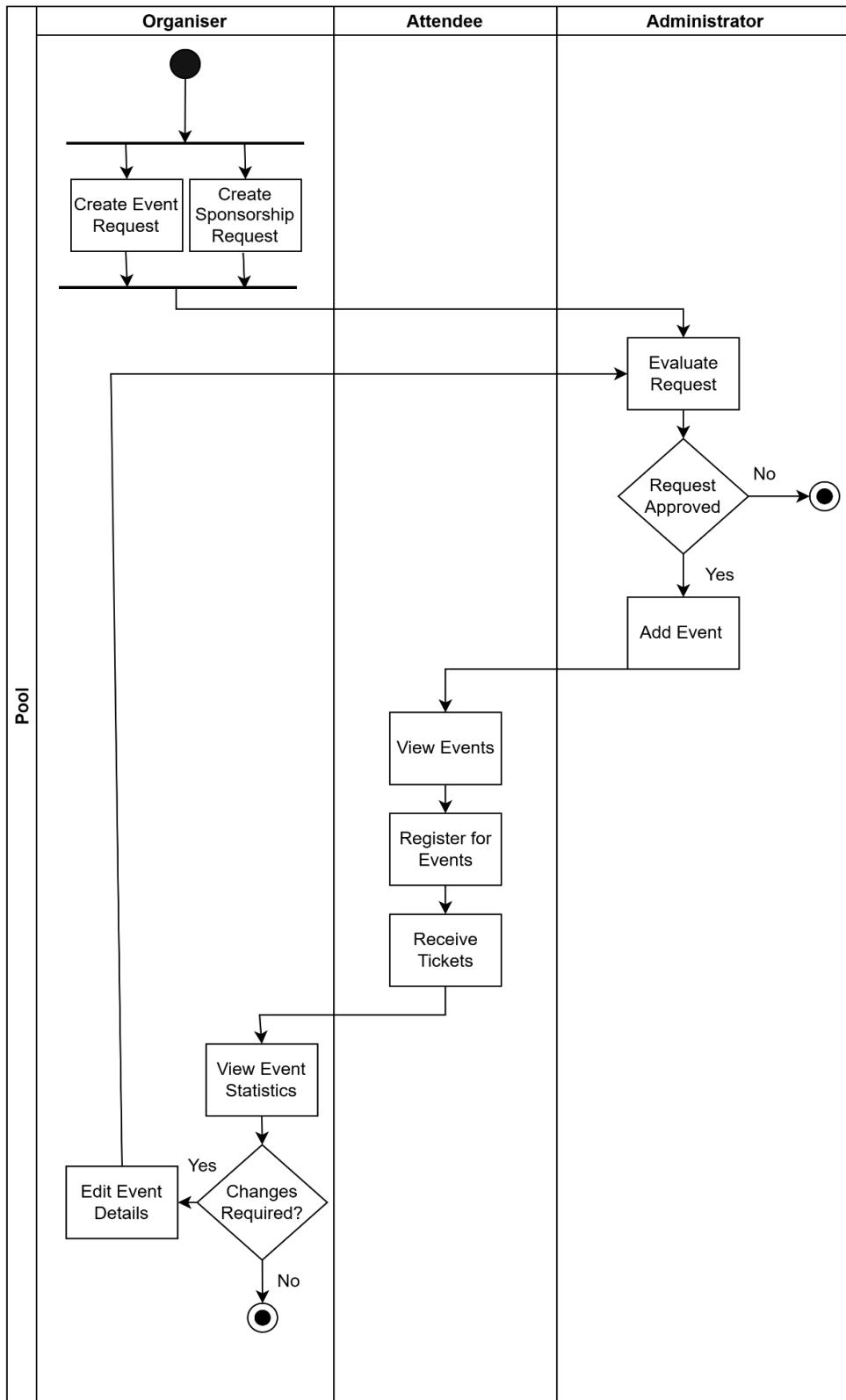
2.1.2 Use Case Template

ID	1
Title	Book an Event
Description	With this facility user can book a ticket for an event which they want to partake in
Primary Actor	User
Preconditions	User must be logged in
Postconditions	User has booked an event
Main Success Scenario	Users have the ability to first view or search for events which have been approved. Once Booked they can view their events.
Owner	TBD

ID	2
Title	Organise an Event
Description	With this facility user can submit the details of their Event and request the administrator
Primary Actor	Organiser
Preconditions	Organiser must be logged in
Postconditions	Organiser has submitted a request
Main Success Scenario	Organisers have the ability to give all the details regarding their event which they can submit to the administrator for approval
Owner	TBD

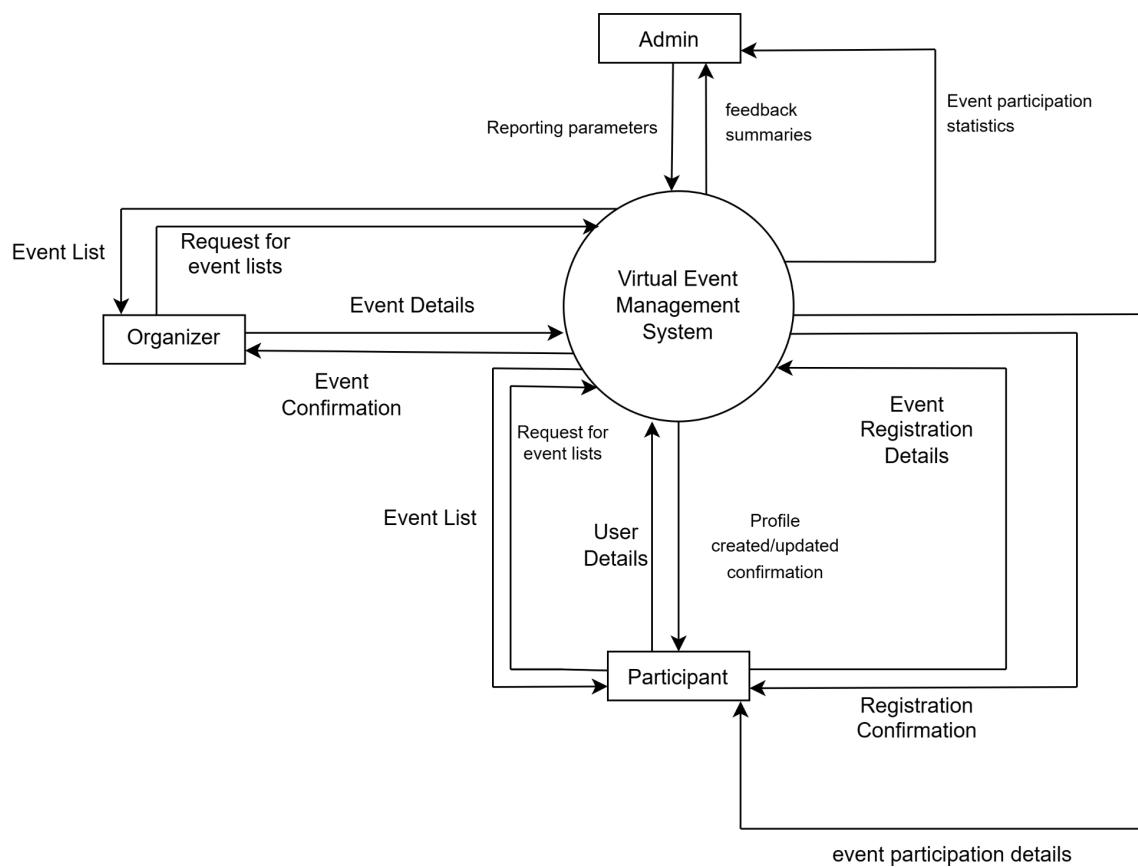
ID	3
Title	View my booked event
Description	With this facility user can view what all events they have registered for
Primary Actor	User
Preconditions	User must be logged in
Postconditions	User has can view all their Events
Main Success Scenario	Users have the ability to easily see what all events they have registered for
Owner	TBD

2.2 Activity Diagrams and Swimlane Diagrams

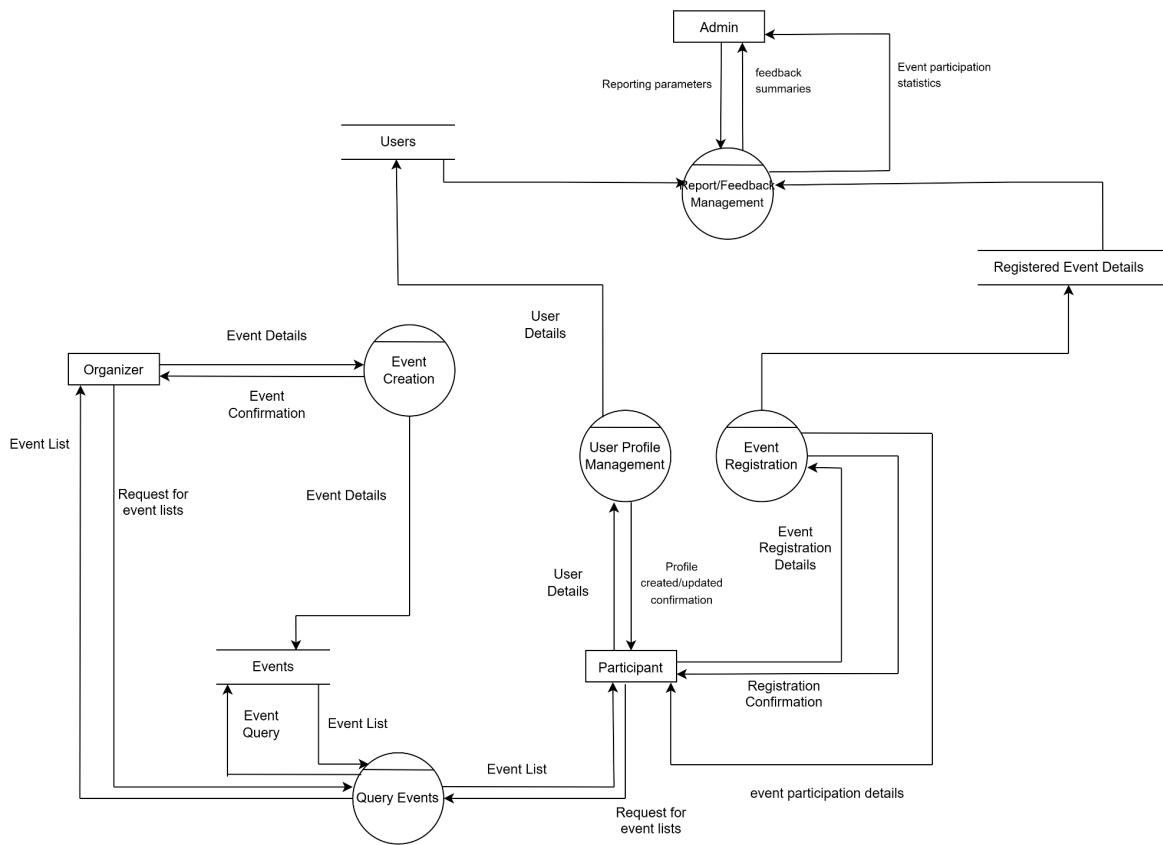


2.3 Data Flow Diagrams

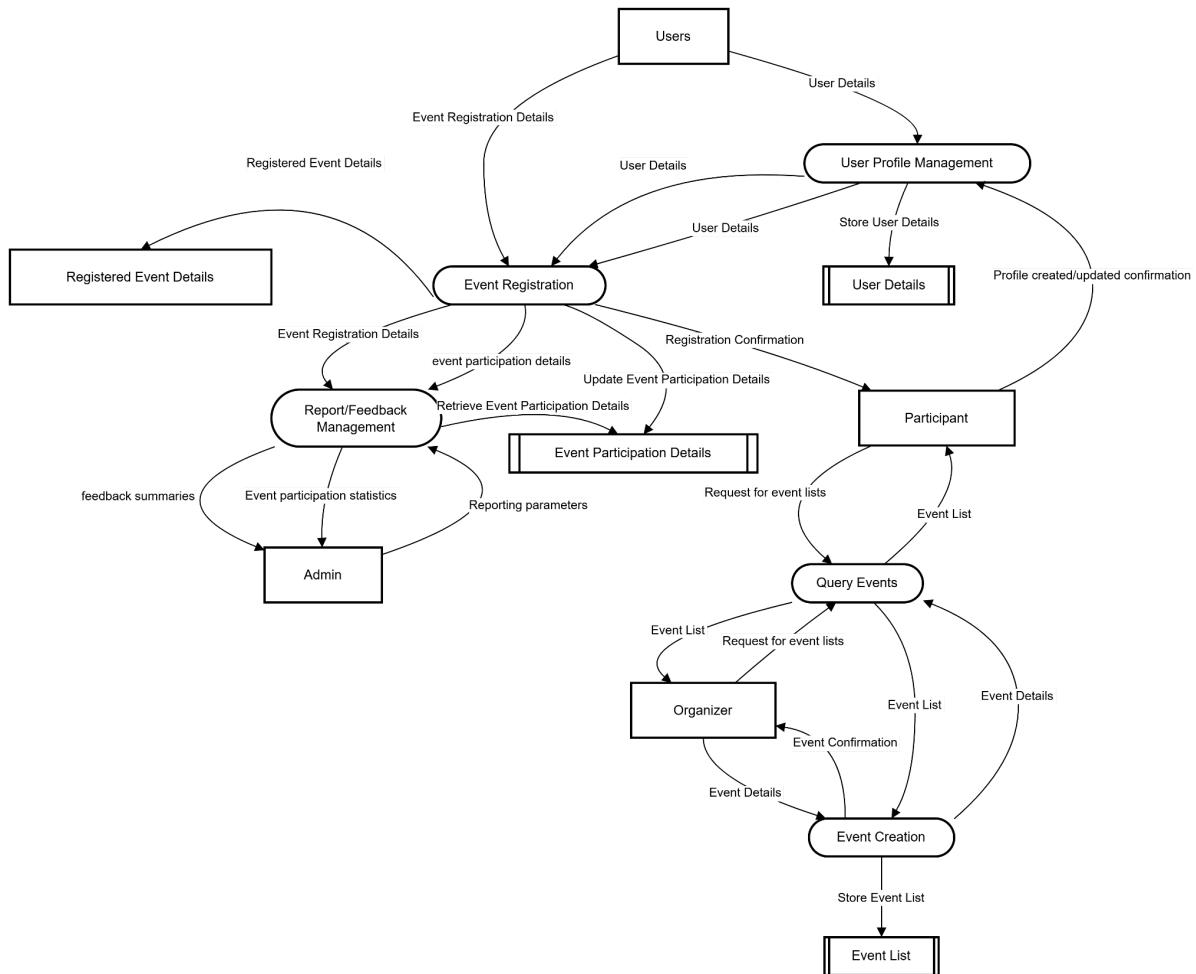
2.3.1 DFD Level 0



2.3.2 DFD Level 1



2.3.2 DFD Level 2



2.4. Software Requirements Specifications in IEEE format

2.4.1 Table of Contents

1. Introduction

- 1.1 Purpose
- 1.2 Scope
- 1.3 Definitions, Acronyms, and Abbreviations
- 1.4 References

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Functions
- 2.3 User Classes and Characteristics
- 2.4 Operating Environment
- 2.5 Design and Implementation Constraints
- 2.6 Assumptions and Dependencies

3. System Features

- 3.1 Event Creation and Management
- 3.2 User Registration and Profile Management
- 3.3 Event Registration
- 3.4 Event Scheduling and Notifications
- 3.5 Reporting and Analytics

4. External Interface Requirements

- 4.1 User Interfaces
- 4.2 Hardware Interfaces
- 4.3 Software Interfaces
- 4.4 Communication Interfaces

5. System Requirements

- 5.1 Functional Requirements
- 5.2 Performance Requirements
- 5.3 Security Requirements
- 5.4 Software Quality Attributes

6. Other Requirements

- 6.1 Legal and Regulatory Requirements
- 6.2 Environmental Requirements

2.4.2 Introduction

1 Purpose

The purpose of this document is to define the requirements for the Event Management System (EMS) specifically tailored for the Thapar Institute of Engineering and Technology (TIET). The system aims to streamline the process of organizing, managing, and tracking events within the institute.

2 Scope

The EMS will support event organizers, participants, and administrators by providing a centralized platform for event creation, registration, management, and reporting. It will cater to various types of events such as workshops, seminars, cultural events, and conferences.

3 Definitions, Acronyms, and Abbreviations

- **EMS:** Event Management System
- **TIET:** Thapar Institute of Engineering and Technology
- **Admin:** System Administrator
- **User:** Event Organizer, Participant

4 References

- IEEE Standard for Software Requirements Specifications (IEEE 830-1998)
- Thapar Institute of Engineering and Technology Website

2.4.3. Overall Description

1 Product Perspective

The EMS will be a web-based application integrated into TIET's existing IT infrastructure. It will be accessible to students, faculty, and staff via their institutional credentials.

2 Product Functions

The main functions of the EMS include:

- Event Creation and Management
- User Registration and Profile Management
- Event Registration and Participation
- Event Scheduling and Notifications
- Reporting and Analytics

3 User Classes and Characteristics

- *Event Organizers*: Faculty or students responsible for organizing events.
- *Participants*: Students, faculty, and external participants attending events.
- *Administrators*: IT staff managing the system and providing support.

4 Operating Environment

- Web-based, accessible via modern browsers (Chrome, Firefox, Edge).
- Backend on a server running Linux/Windows, with a database (e.g., MySQL, PostgreSQL).
- Integration with TIET's authentication systems (e.g., LDAP, SSO).

5 Design and Implementation Constraints

- The system must comply with TIET's IT security and privacy policies.

- The system should be scalable to handle peak loads during large events.

6 Assumptions and Dependencies

- Users will have internet access and basic computing skills.
- The institute's IT department will provide necessary infrastructure support.

2.4.4. System Features

1 Event Creation and Management

- **Description:** Event organizers can create and manage events.
- **Inputs:** Event details (title, date, time, venue, description).
- **Processing:** Store event details, validate inputs, notify stakeholders.
- **Outputs:** Confirmation of event creation, listing on the events page.

2 User Registration and Profile Management

- **Description:** Users can register on the EMS and manage their profiles.
- **Inputs:** User details (name, email, role, contact information).
- **Processing:** Validate user details, create or update profiles.
- **Outputs:** User profile creation or update confirmation.

3 Event Registration

- **Description:** Users can register for events.
- **Inputs:** User ID, event ID.
- **Processing:** Check event availability, register users, send confirmation.
- **Outputs:** Registration confirmation, event participation details.

4 Event Scheduling and Notifications

- **Description:** The system schedules events and sends notifications.
- **Inputs:** Event schedule details, user notification preferences.
- **Processing:** Generate calendar entries, send email/SMS notifications.
- **Outputs:** Event reminders, schedule updates.

5 Reporting and Analytics

- **Description:** Admins and organizers can generate reports.
- **Inputs:** Reporting parameters (event type, date range, user demographics).
- **Processing:** Generate reports based on database queries.
- **Outputs:** Event participation statistics, feedback summaries.

2.4.5. External Interface Requirements

1 User Interfaces

- A responsive web interface for all user roles.
- An admin dashboard for system management.

2 Hardware Interfaces

- Servers with sufficient storage and processing power to handle concurrent users.

3 Software Interfaces

- Integration with TIET's authentication system (e.g., LDAP).
- Email and SMS gateways for notifications.

4 Communication Interfaces

- Secure HTTPS protocol for all communications.
- WebSocket or similar technology for real-time updates.

2.4.6. System Requirements

1 Functional Requirements

- User Authentication
- Event Management
- Registration and Ticketing
- Notifications and Alerts
- Reporting and Analytics

2 Performance Requirements

- The system should support up to 10,000 concurrent users.
- Response time for any action should be under 2 seconds.

3 Security Requirements

- Role-based access control
- Data encryption in transit and at rest
- Regular security audits and updates

4 Software Quality Attributes

- Usability: The system should be easy to use with minimal training.
- Reliability: System uptime should be 99.9%.
- Scalability: The system should scale to handle increasing users and events.
- Maintainability: The system should be modular and easy to update.

2.4.7. Other Requirements

1 Legal and Regulatory Requirements

- **Compliance with Data Protection Laws:** The EMS must comply with relevant data protection and privacy regulations, ensuring that personal data is handled securely and only for the intended purposes.

2 Environmental Requirements

- **Energy Efficiency:** The EMS should be optimized for energy efficiency, especially if deployed on-premises, to minimize power consumption.
- **Sustainability:** The development and deployment of the EMS should consider sustainability practices, such as utilizing cloud services that adhere to green computing principles.

2.5 User Stories and Story Cards

Front of Story Card:

Title: EMS - Simplifying Event Management at TIET

User Story:

As an event organizer, I want to create, manage, and promote events seamlessly using a centralized system. With EMS, I can streamline the event creation process, register participants, schedule notifications, and generate insightful reports. This ensures efficient event management, improved participation, and valuable data analytics for future improvements.

Back of Story Card:

Confirmation

Success Scenario:

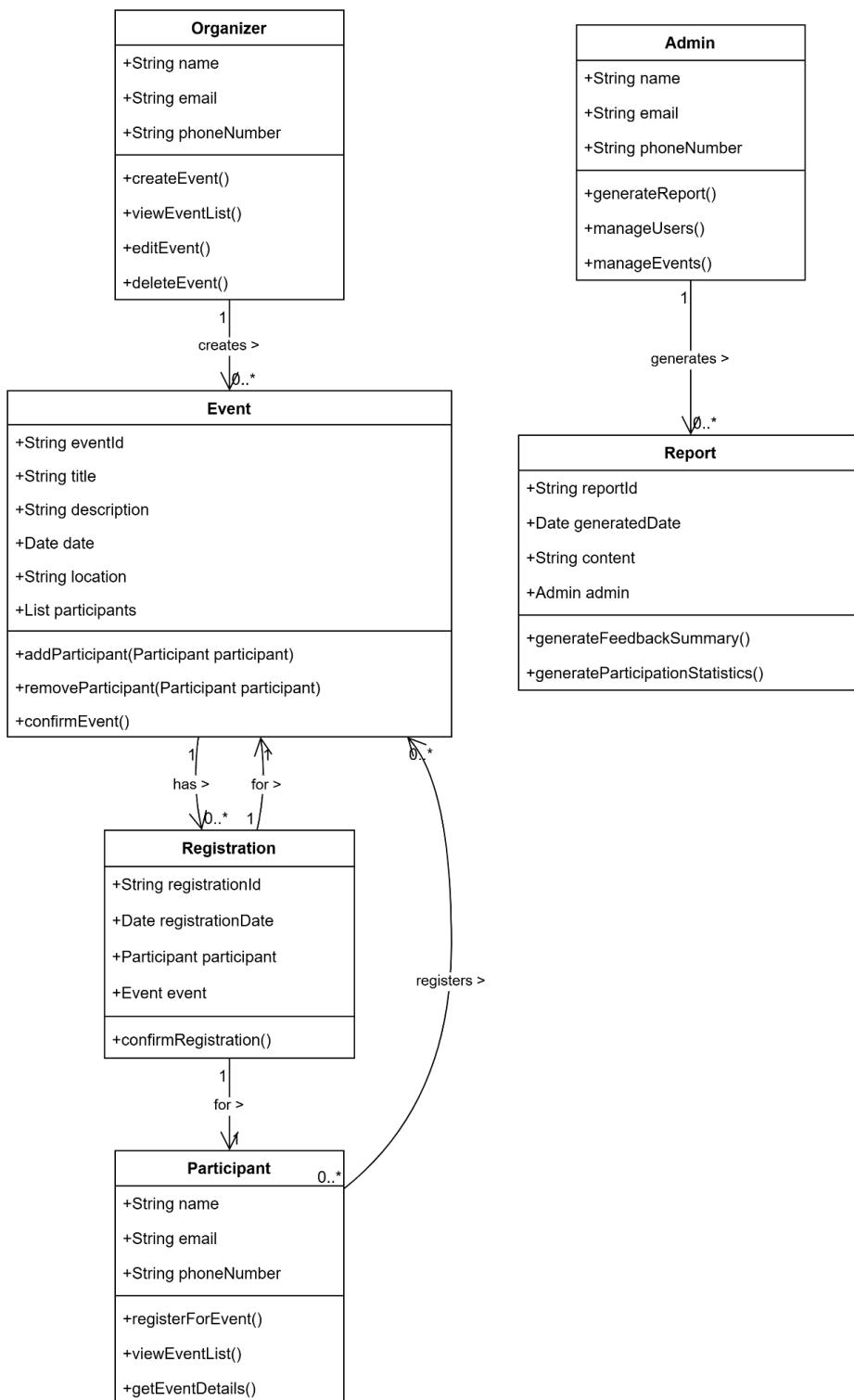
- The organizer logs into the EMS using institutional credentials.
- The organizer creates an event by filling in details (title, description, date, time, venue).
- EMS validates the input, stores the event data, and displays a confirmation.
- Notifications about the event are sent to the targeted audience (students, faculty).
- Participants register successfully, and their details are updated in the system.
- The event is conducted successfully, and feedback is collected for reporting and analytics.
- The organizer generates and downloads participation statistics and feedback summaries.

Failure Scenario:

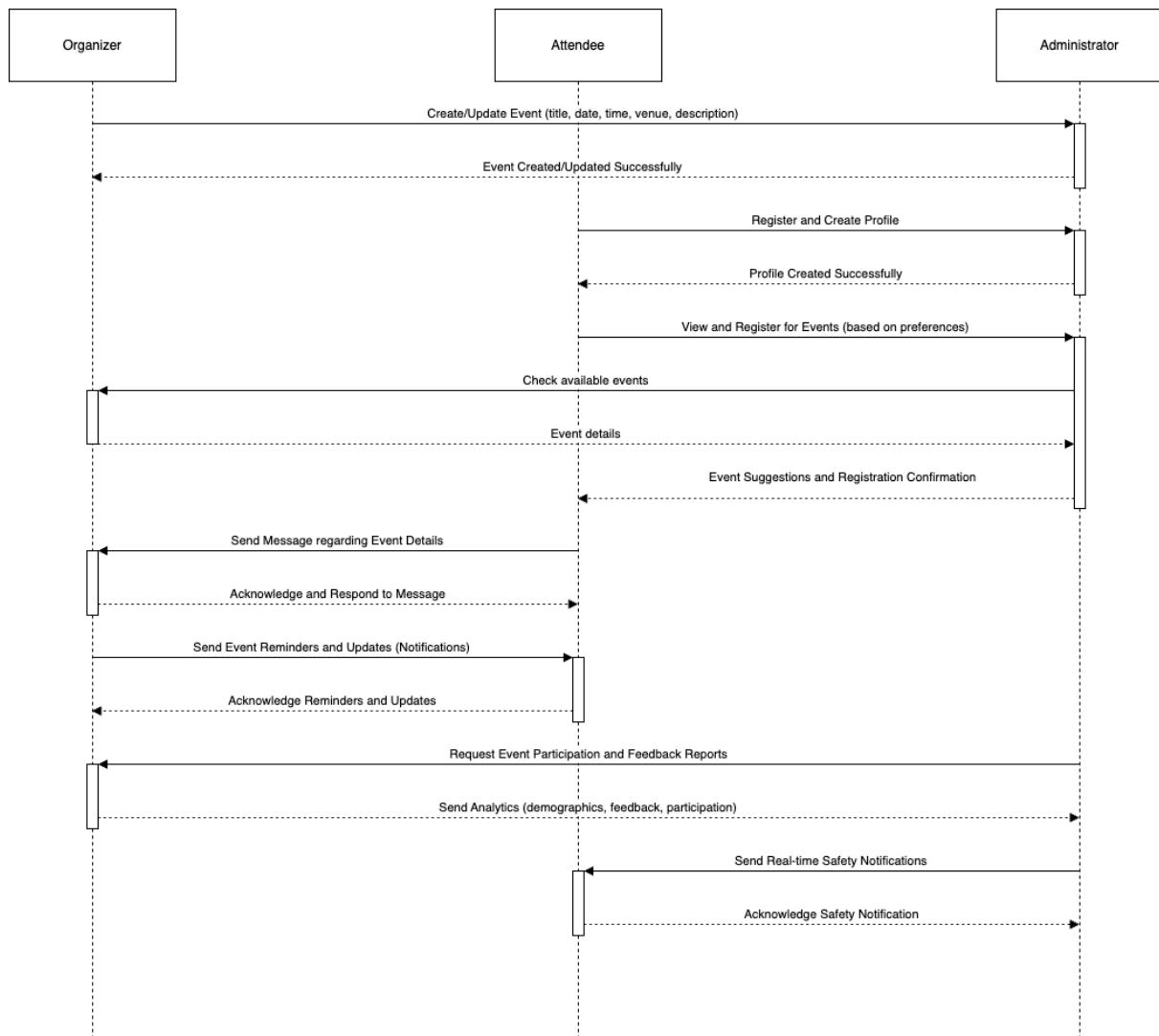
- The organizer provides incomplete or invalid event details, and EMS prompts for corrections.
- Notifications fail to reach the audience due to incorrect email/SMS settings.
- Participants are unable to register due to server downtime or technical issues.
- Reports are not generated due to missing or incomplete event data.
- Scalability issues during a large event lead to delayed processing or system crash.

6. Design Phase (At least two significant cases of each diagram)

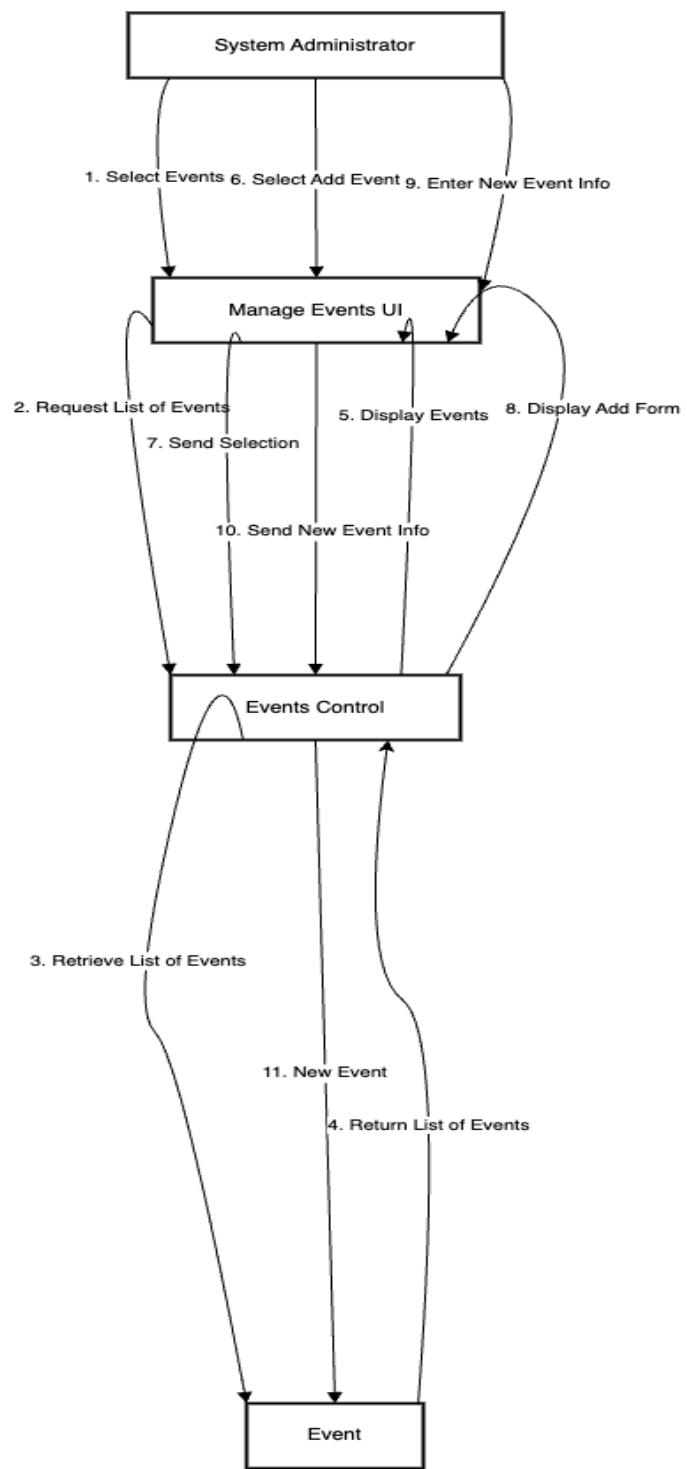
3.1 Class Diagram



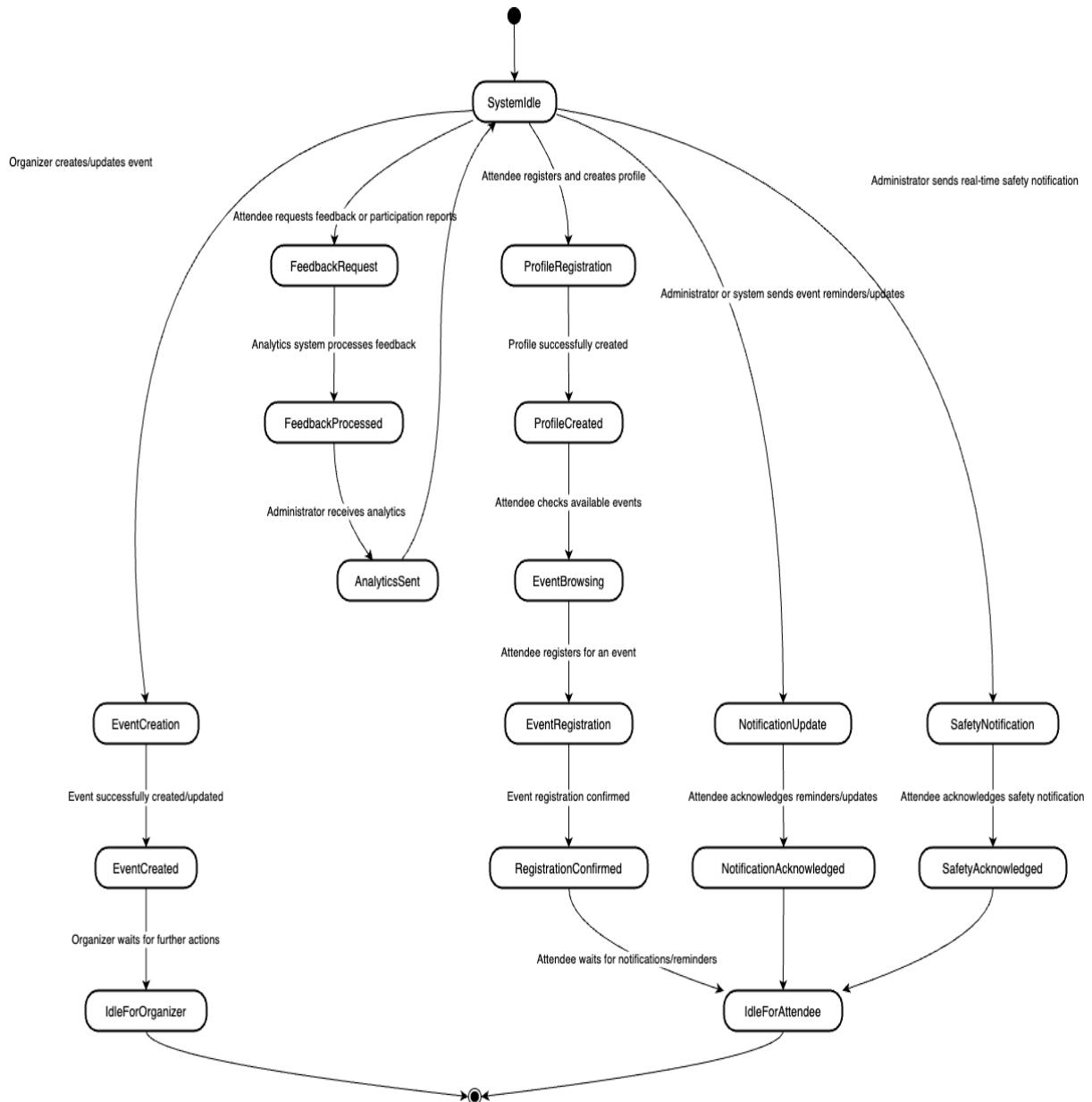
3.2 Sequence Diagram



3.3 Collaboration Diagram

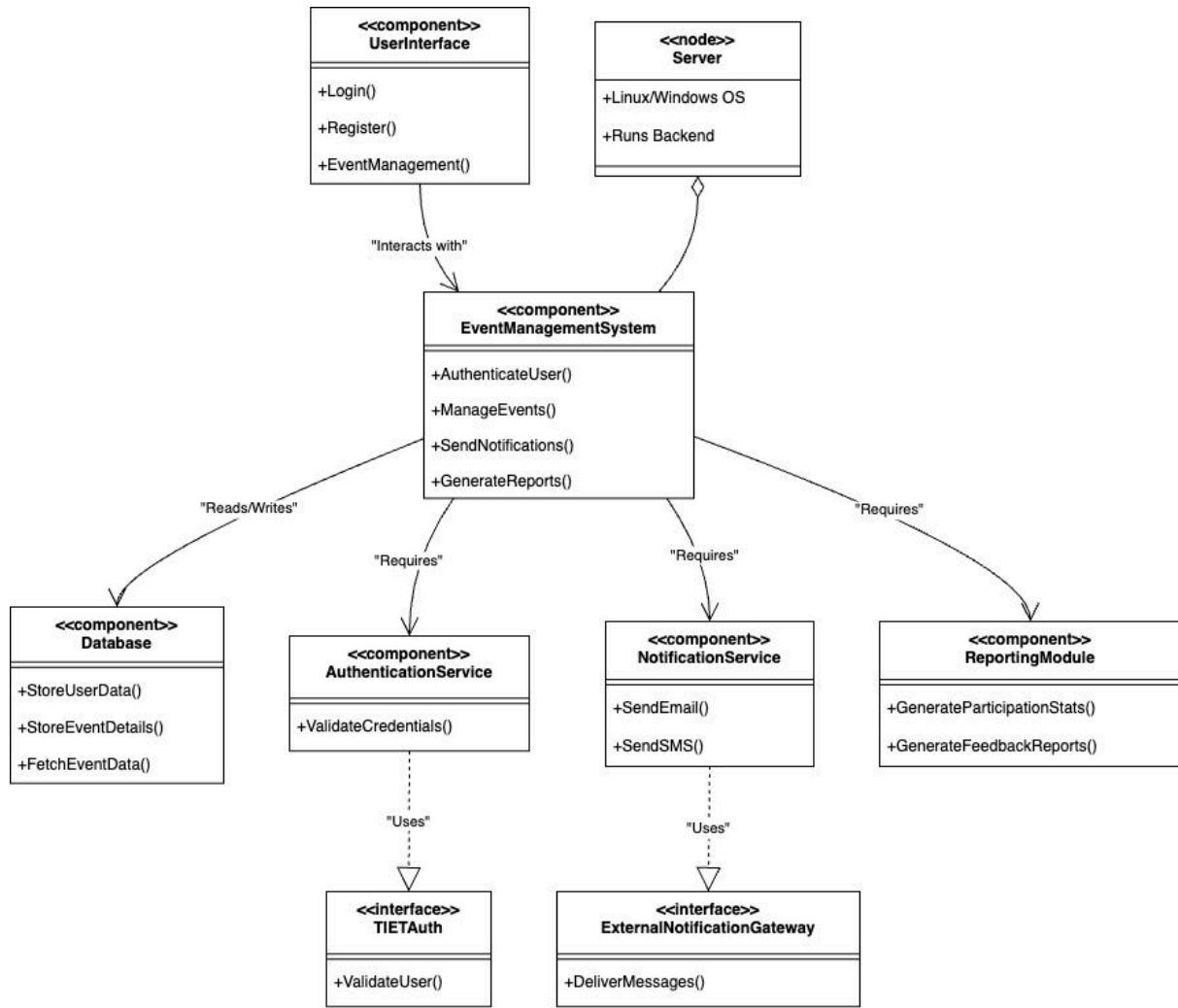


3.4 State Chart Diagrams

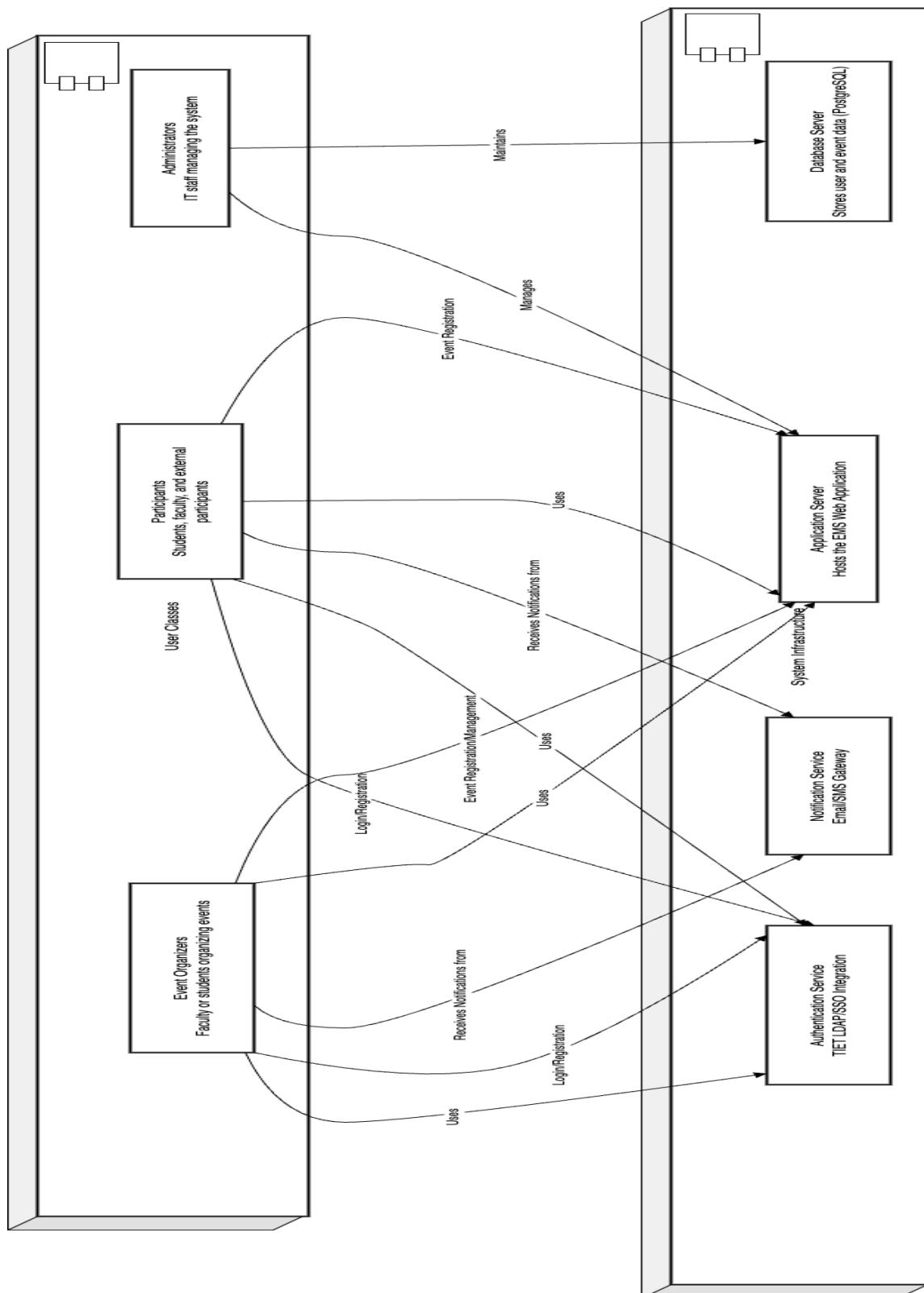


6. Implementation

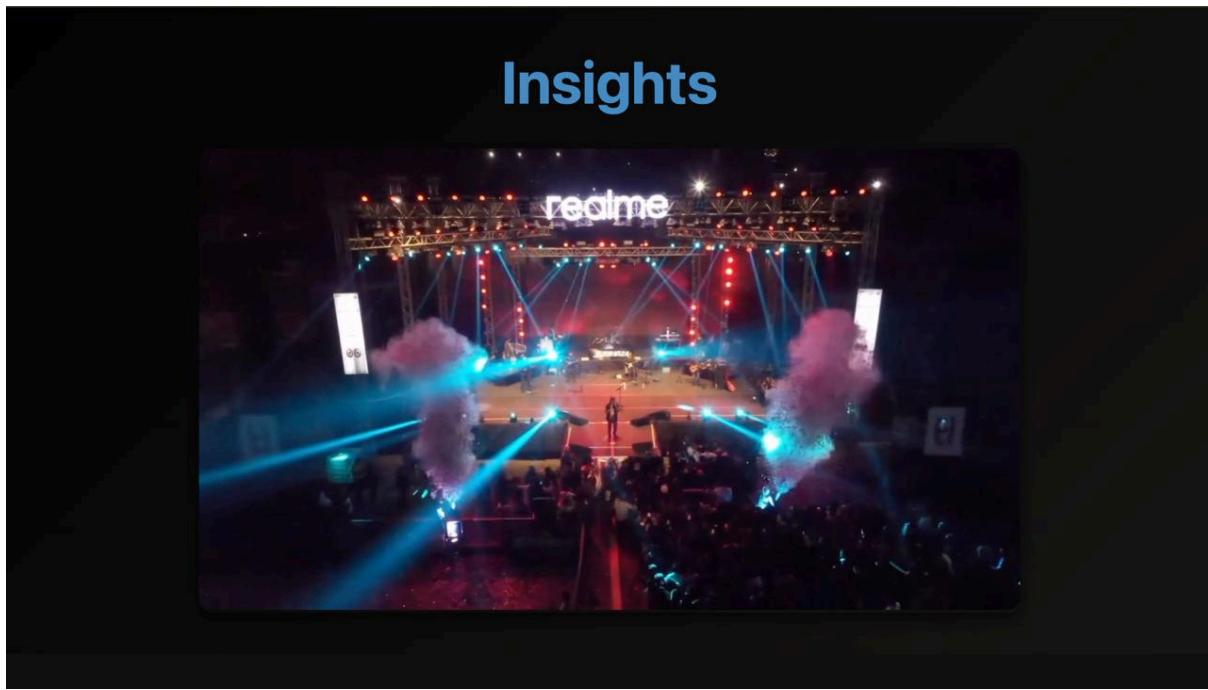
4.1 Component Diagrams



4.2 Deployment Diagrams



4.3 Screenshots



Our Sponsors



Our Team

Dhruv Singh
Pundir

Dhruv Singh
Pundir
Tech Lead

Shaurya Jain

Tech Lead

Guransh Singh

Tech Lead

Arshdeep Aswal

Tech Lead

Registration

X

Enter your name(optional)

Enter your email

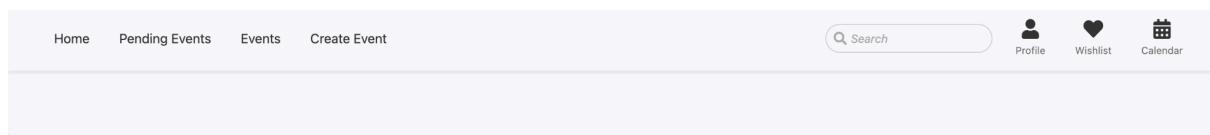
Create password

Confirm password

I accept all terms & conditions

Register Now

Already have an account? [Login now](#)



Create Event

Event Name**Description****Location****Start Time** **End Time** **Create Event**

Societies

Explore various societies at Thapar University

GAMES AND SPORTS

The University has several playgrounds and well-maintained athletic track to encourage the students to take part in different games such as Cricket, Hockey, Football, Basketball, Volleyball, Lawn Tennis and Badminton. The University has also a Gymnasium-cum-Badminton Hall and a Swimming Pool Complex equipped with all modern facilities. The sports department organizes various sporting events like 'Thaparympics', 'URJA' etc.

MUSIC AND DRAMATIC SOCIETY (MUDRA)

The objective of this society is to hone the extra-curricular skills of students in the area of Music, Dramatics and developing managerial prowess contributing towards their overall personality. It organizes several big events such as MUDRA night and Izhaar etc.

CREATIVE COMPUTING SOCIETY (CCS)

CCS society basic objective is to encourage students to actively participate in computer activities other than their curriculum. Activities: Chakravayu, Compu Quiz contests, Algorithm mania, Web Hunt, Programming – Overnight & Long term

DANCE CLUB 'NOX'

To hone the dancing and social skills of students, that contributes to their overall personality development. The club organizes various dance workshops on campus and students of this club actively participate in various cultural festivals across institutes.

PARYAVARAN WELFARE SOCIETY (PWS)

The PWS was set up in the year 2009, as a scheme to uphold biodiversity, guard "paryavaran" (environment), and save energy resources at local level. This society also spearheads with the 'Swachh Bharat' campaign on the TU campus.

ADVENTURE CLUB

Adventure Club @ Thapar University is the forum for adventure enthusiasts, trekkers, climbers and armchair mountaineers in the student community. It works to develop the spirit of adventure in Thaparians.

FROSH

FROSH WEEK comprises of various activities and events for the fresher class of Thapar University to ease them into their new surroundings

ECONOMICS CLUB (ECON CLUB)

Econ Club at Thapar University is dedicated towards inculcating economics and financial know-how among the student community.



ECON



ADVENTURE CLUB



Mudra
MUSIC DANCE DRAMATICS

THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY

P.O. Box 32, Patiala, Punjab 147004, India

QUICK LINKS

- [Home](#)
- [Events](#)

CONTACT US

[@](#)
[Email](#)
[Website](#)

 Phone: +91-175-2393021

Connect with Our Sponsors

**Realme**

Leading in technology solutions.

**Maruti Suzuki**

Innovating the future.

**Subway**

Purveyor of healthier fast food.

**RedBull**

Energize Your Every Moment

**Shiv Naresh**

Precision in Every Stitch



5. Testing Phase

5.1 Test Plan

5.1.1 Test Objectives

The objective of the test is to verify that the functionality of the **Event Management System** works according to the specifications. The test will focus on:

- Executing and verifying the authentication process.
- Identifying, fixing, and retesting all high and medium severity defects while considering all constraints.

Key Points

- Production-like data will be available in the system prior to starting functional testing.
 - Exploratory testing will be carried out once the build is ready for testing.
 - Performance testing is **not** included in this phase.
 - All defects must include a snapshot in JPEG format.
 - Functional testing will include verifying error-free data handling and login API delivery.
-

5.1.2 Test Principles

- Testing processes will be well-defined yet flexible, allowing adjustments as needed.
 - Testing will be repeatable, quantifiable, and measurable.
 - Testing will be divided into distinct phases, each with clearly defined objectives and goals.
-

5.1.3 Data Approach

- In functional testing, a sample data set will be used to evaluate data handling functionality.
-

5.1.4 Scope and Levels of Testing

Purpose:

Ensure critical defects are removed before progressing to subsequent testing levels.

Scope:

Focus on first-level navigation.

Testers:

Software developers.

Method:

Exploratory testing, performed without test scripts or documentation, using only predefined test cases.

5.2 Test Cases

5.2.1 Test Strategies

To deliver bug-free software, testers aim to cover as many conditions as possible. However, exhaustive testing is impractical due to time and resource constraints. Instead, the following strategies will be applied:

5.2.1.1 Equivalence Class Partitioning

- Divides input data into equivalence classes.
- Test cases are designed for each class to reduce time and optimize testing.

5.2.1.2 Boundary Value Analysis

- Focuses on boundary values for each equivalence partition.
- Includes both valid and invalid boundary values, often used in stress and negative testing.

5.2.2 Testing cases

Test Cases: performed while developing the ‘Lost and Found’ flutter Application:

- **Test Case 1 :** On entering the incorrect username/password while logging into the system, an error message is displayed saying “ Invalid username/password. Please try again”

Test Case No.: 1 Attempt System : Team TVEMS Designed By : Team TVEMS Executed By: Guransh Singh Short Description : Testing of user login	Test Case Name : Invalid Login Subsystem : Login Design Date : 5/11/2024 Execution Date : 12/11/2024
--	---

Pre-Condition

- Customers should have a valid email and password.

Step	Action	Expected System Response	pass/fail	Comments
1.	Enter Email	Email must be valid	Pass	
2.	Enter Password	Password must be valid	Pass	
3.	Press “Login” Button	Software should show error message of incorrect credentials:	Pass	
4.	Check post-condition 1	The user should remain on the login page without being logged in. A clear error message should be displayed.	Pass	

Post-Condition:

An error message, “Error: Invalid Credentials. Please Try Again.” is displayed asking the user to enter username and password again.

- **Test Case 2 :**Validate that a user cannot register for an event they have already registered for.

Test Case No.: 2	Test Case Name : user tries to mistakenly register again
Attempt System : TVEMS	Subsystem : Event
Designed By : Team TVEMS	Design Date : 5/11/2024
Executed By: Arshdeep Aswal	Execution Date : 12/11/2024
Short Description : Prevent Duplicate Event Registration	

Pre-Condition

- Customer should be logged in.

Step	Action	Expected System Response	pass/fail	Comments
1.	Navigate to the event details page for an event they have already registered for.	The system prevents duplicate registration and displays a message: " <i>You have already registered for this event.</i> "	Pass	
2.	Attempt to register for the event again by clicking the "Register" button.	The registration process is not repeated.	Pass	

Post-Condition:

Could not re-register for the event

- **Test Case 3 :** Validate that a logged-in user can successfully submit a request to create an event, and the request is marked as pending for admin review.

Test Case No.: 3 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Shaurya Jain Short Description : user gives request to create event	Test Case Name : Create Event Subsystem : Create Event Design Date : 5/11/2024 Execution Date : 12/11/2024			
Pre-Condition				
	<ul style="list-style-type: none"> • The user must be logged into the system. • The "Create Event" feature should be enabled in the system. 			
Step	Action	Expected System Response	pass/fail	Comments
1.	Navigate to the "Create Event" page.	The event details are validated for completeness and correctness.	Pass	
2.	Fill in all required event details (e.g., event name, description, date, time, venue).	The event creation request is saved to the database with a status of "Pending"	Pass	
3.	Submit the event creation request by clicking the "Submit" button.	A confirmation message is displayed to the user: <i>"Your event request has been submitted and is pending approval by the admin."</i>	Pass	
Post-Condition:				
The event request is stored in the database with a status of "Pending Approval." The admin panel should now display the submitted event request in the "Pending Requests" section.				

- **Test Case 4 :** User tries to book a registration without logging in

Test Case No.: 4 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Dhruv Singh Short Description : User tries to logout of the system	Test Case Name : Search Events Subsystem : Events Design Date : 5/11/2024 Execution Date : 12/11/2024
--	--

Pre-Condition

- The user must not be logged in the system
-

Step	Action	Expected System Response	pass/fail	Comments
1.	Navigate to the "Events" page	We get a list of events with short descriptions and timings	Pass	
2.	Click on the register button	The system tells the user that they are not authorised and need to log in	Pass	
3.	Click the "Search" button	The user can view event details from the search results.	Pass	

Post-Condition:

An error message is displayed which tells the user the need to login to register.

- **Test Case 5 :** Validate that a logged-in user can view a list of events they have registered for.

Test Case No.: 5 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Guransh Singh	Test Case Name : View Registered Events Subsystem : Registered events Design Date : 5/11/2024 Execution Date : 12/11/2024			
Short Description : user tries to view their registered events				
Pre-Condition				
<ul style="list-style-type: none"> ● The user must have a valid student account. ● The user must be logged into the system. ● The system should have at least one event registered by the user. 				
Step	Action	Expected System Response	pass/fail	Comments
1.	Log in as a student using valid credentials.	The system retrieves and displays a list of events the user has registered for.	Pass	
2.	Navigate to the "My Registered Events" section.	Each event in the list includes details such as event name, date, time, and venue.	Pass	
3.	Review the displayed list of registered events.	If the user has not registered for any events, the message " <i>You have not registered for any events yet</i> " is displayed.	Pass	
Post-Condition:				
An error message, “Error: Invalid Credentials. Please Try Again.” is displayed asking the user to enter username and password again.				

- **Test Case 6 :** Validate that any user (logged in or not) can view the list of all events available in the system.

Test Case No.: 6 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Arshdeep Aswal Short Description : user wants to see all events	Test Case Name : View All Events Subsystem : Events Design Date : 5/11/2024 Execution Date : 12/11/2024
---	--

Pre-Condition

- The system should have events stored in the database.
- The "View All Events" feature should be enabled and accessible.
- The user may or may not be logged in.

Step	Action	Expected System Response	pass/fail	Comments
1.	Navigate to the "All Events" page or section of the system.	The system retrieves and displays a list of all available events in chronological or categorized order.	Pass	
2.	scroll through the list of events displayed.	Each event entry includes key details such as event name, date, time, venue, and description.	Pass	

Post-Condition:

The user is presented with a complete list of events stored in the system.

- **Test Case 7 :** Validate that an admin can view a list of all event requests pending approval.

Test Case No.: 7 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Guransh Singh Short Description : admin wants to see all pending events	Test Case Name : View Pending Event Requests Subsystem : Admin Design Date : 5/11/2024 Execution Date : 12/11/2024
---	---

Pre-Condition

- The admin must have a valid account with administrative privileges.
- The admin must be logged into the system.
- There should be at least one event request in the system with a "Pending Approval" status.

Step	Action	Expected System Response	pass/fail	Comments
1.	Log in as an admin using valid credentials.	The system retrieves and displays a list of all events with a status of "Pending Approval."	Pass	
2.	Navigate to the "Pending Event Requests" section in the admin dashboard.	Each event entry includes key details such as event name, requested by (user name or ID), date, time, and description.	Pass	
3.	Review the displayed list of pending event requests.	If there are no pending event requests, the message " <i>No pending event requests at the moment</i> " is displayed.	Pass	

Post-Condition:

The admin is presented with an accurate list of pending event requests.
The system logs the admin's action for audit purposes.

- **Test Case 8 :** Validate that non-admin users cannot access the admin page or its features.

Test Case No.: 8	Test Case Name : Admin Page Access Restriction
Attempt System : TVEMS	Subsystem : Login
Designed By : Team TVEMS	Design Date : 5/11/2024
Executed By: Guransh Singh	Execution Date : 12/11/2024
Short Description : average user tries to see admin page	

Pre-Condition

- The user must have a valid account but without administrative privileges.
- The user must be logged into the system.
- The system must have role-based access control implemented.

Step	Action	Expected System Response	pass/fail	Comments
1.	Log in as a regular user (non-admin) using valid credentials.	The system denies access to the admin page for non-admin users.	Pass	
2.	Attempt to navigate to the admin page via URL manipulation or the user interface.	An appropriate error message is displayed, such as " <i>Access Denied: You do not have permission to view this page.</i> "	Pass	

Post-Condition:

The user remains restricted from accessing the admin page.
The system logs the unauthorized access attempt for security purposes.

- **Test Case 9 :** Validate that an admin can approve or decline pending event requests.

Test Case No.: 9 Attempt System : TVEMS Designed By : Team TVEMS Executed By: Shaurya Jain Short Description : Testing of user login	Test Case Name :Approve/Decline Pending Event Requests Subsystem : Pending Requests Design Date : 5/11/2024 Execution Date : 12/11/2024
--	--

Pre-Condition

- The admin must have a valid account with administrative privileges.
- The admin must be logged into the system.
- There should be at least one event request in the system with a status of "Pending Approval."

Step	Action	Expected System Response	pass/fail	Comments
1.	Log in as an admin using valid credentials.	If the admin clicks "Approve," the event status is updated to "Approved," and the event becomes visible to all users.	Pass	
2.	Navigate to the "Pending Event Requests" section in the admin dashboard.	If the admin clicks "Decline," the event status is updated to "Declined," and the event is not displayed to users.	Pass	
3.	Select an event from the list of pending requests.	The system displays a success message, such as " <i>Event approved successfully</i> " or " <i>Event declined successfully</i> ."	Pass	
4.	Click on the "Approve" button to approve the event or the "Decline" button to reject the event.	If a reason for declining is provided, it is stored in the database and visible to the user who created the event.	Pass	

Post-Condition: event status updated in database

- **Test Case 10 :** Validate that an event must be visible if the administrator has accepted it and not if they have declined

Test Case No.: 10	Test Case Name : notify user of admin reply
Attempt System : TVEMS	Subsystem : Admin Page
Designed By : Team TVEMS	Design Date : 5/11/2024
Executed By: Dhruv Singh	Execution Date : 12/11/2024
Short Description : Event starts to be visible after approval/not after decline	

Pre-Condition

- The user must have submitted a pending event request.
- An admin must log in and approve or decline the event.

Step	Action	Expected System Response	pass/fail	Comments
2.	navigate to the "Pending Event Requests" section.	The admin accepts or declines the request	Pass	
3.	Approve or decline the pending event.	If admin accepts then the event is visible to everyone to register for	Pass	

Post-Condition:

The response is stored successfully.