External Events Management Portal System Requirements Specification (SRS)

EME-PORTAL

NAME:	SABARISH K S
REG NO. :	7376222AD189
PROJECT ID :	13
MODULE NAME :	EXTERNAL EVENT MANAGEMENT SYSTEM

1. Introduction

This document outlines the System Requirements Specification (SRS) for the External Events Management Portal, a web application designed to facilitate the submission, approval, and management of external event proposals submitted by students and faculty members at [Office Special Labs].

2. System Overview

The External Events Management Portal is a web-based application that streamlines the process of managing external event proposals. It provides functionalities for students, faculty, Head of Office, and administrators to manage the entire lifecycle of an event proposal, from submission to approval and post-event tracking.

3. System Features

- **User Management:**
    - Supports user roles: Student, Faculty, Head, Admin
    - Admin manages user accounts (create, edit, delete)
- **Event Management:**
    - Students and Faculty can submit event proposals

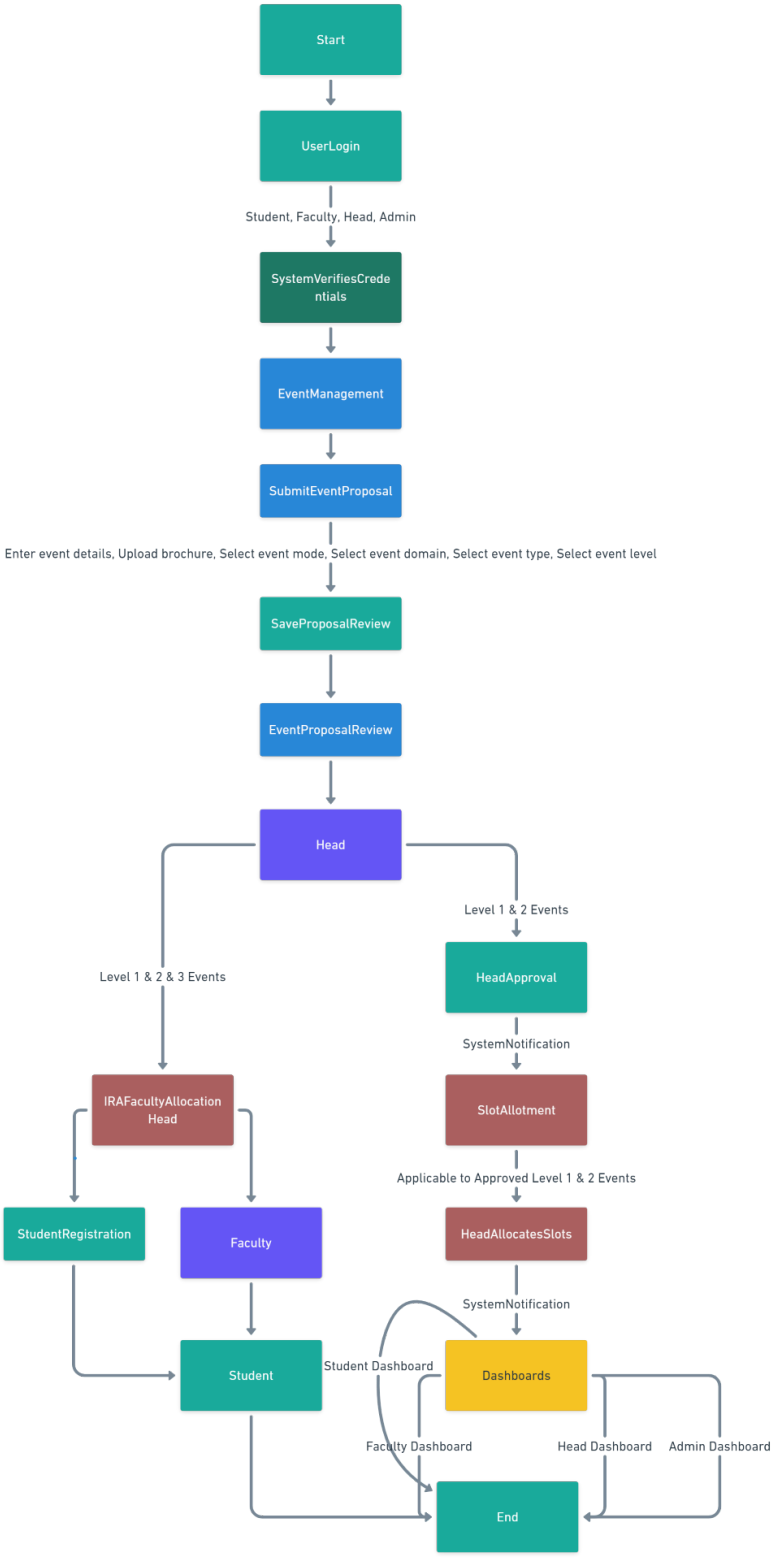
    - Proposal details include:
        - Event Name
        - Organizer
        - Date
        - Registration Deadline
        - Location
        - Link to event website (optional)
        - Brochure upload (PDF)
        - Mode (Online/Offline)
        - Domain (Hardware/Software)
        - Type (Technical Paper/IEEE/Project/Workshop/Other)
        - Level (National/International)
    - Head categorizes events based on approval level (Level 1, Level 2, Level 3)
    - Head assigns faculty to conduct Internal Review Assessment (IRA) for Level 1 and Level 2 events
- **Internal Review Assessment (IRA):**
    - Students can register for IRA events before 12 hours of the scheduled date
    - Assigned Faculty reviews registered students and assigns marks
    - Head can view details of students who passed the IRA
- **Event Approval:**
    - Head approves/rejects event proposals and categories (Level 1, Level 2,level 3)
    - System notifies applicant (student) and all students about approval status
- **Slot Allotment (Level 1 & 2 Events):**
    - Head allocates slots for approved events based on available venue capacity
    - System notifies registered students about slot allocation
- **Dashboards:**
    - Admin Dashboard: View list of events, IRA details, and registered students (with edit access)
    - Head Dashboard: View similar details as Admin Dashboard (with some overlap with Student Dashboard)

    - Student Dashboard: View list of approved events, registered events, and IRA results (if applicable)
- **Reporting:**
    - Admin can generate reports on completed events, including students qualified, faculty involved, and IRA dates

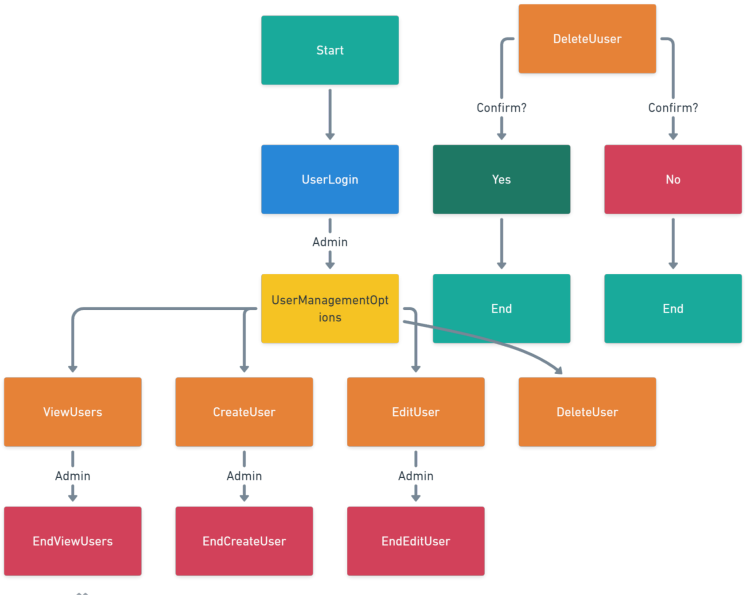
4. Functional Requirements

- **User Registration and Login:** Users can register and log in to the system using their unique credentials.
- **Event Proposal Submission:** Students and Faculty can submit new event proposals with detailed information.
- **Event Proposal Review:** Head reviews proposals based on predefined criteria and categorizes them for approval levels.
- **Internal Review Assessment (IRA) Management:**
    - Head assigns faculty to conduct IRA for designated events.
    - Faculty review registered students and assign marks.
    - Students can view their IRA results.
- **Event Approval Workflow:** Head/Admin approve/reject event proposals based on category.
    - System notifies relevant users about approval status.
- **Slot Allotment:** Head allocates slots for approved events based on venue capacity.
    - System notifies registered students about slot allocation.
- **Dashboard Access:** Users can access relevant dashboards based on their roles.
- **Reporting:** Admin can generate reports on completed events

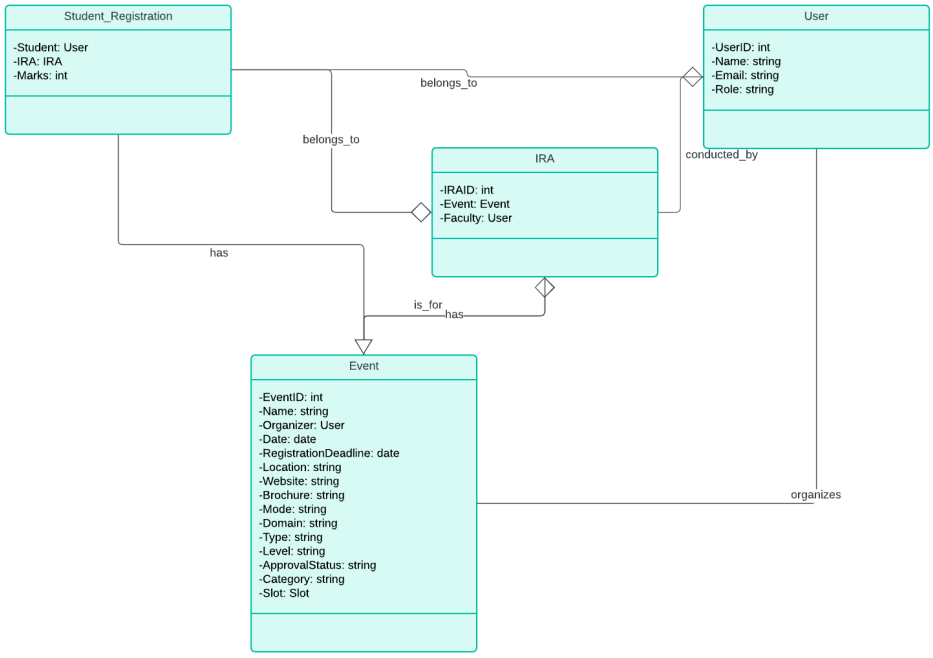
Flow chart :


Start
UserLogin
Student, Faculty, Head, Admin
SystemVerifiesCredentials
EventManagement
SubmitEventProposal
Enter event details, Upload brochure, Select event mode, Select event domain, Select event type, Select event level
SaveProposalReview
EventProposalReview
Head
Level 1 & 2 & 3 Events
Level 1 & 2 Events
HeadApproval
SystemNotification
IRAFacultyAllocation Head
SlotAllotment
Applicable to Approved Level 1 & 2 Events
StudentRegistration
Faculty
HeadAllocatesSlots
SystemNotification
Student
Student Dashboard
Dashboards
Faculty Dashboard
Head Dashboard
Admin Dashboard
End

User management flow chart :



ER DIAGRAM :



5. Non-Functional Requirements

- **Performance:** The system should have acceptable response times for all functionalities.
- **Security:** The system should implement user authentication and authorization mechanisms to restrict access to sensitive data.
- **Scalability:** The system should be scalable to accommodate an increasing number of users and events.
- **Availability:** The system should be highly available with minimal downtime.
- **Usability:** The user interface should be intuitive and user-friendly for all user roles.

6. Technical Specifications

- **Front-End:** HTML, CSS, JavaScript
- **Back-End:** Python, Django Framework
- **Database:** PostgreSQL or MySQL
- **API:** OpenAPI (RESTful API)

7. Appendix

- Data Flow Diagrams (optional)

Technical Specifications & User Interface Mockups

1.1 Technologies

- **Front-End:**
    - HTML5 for structuring web page content
    - CSS3 for styling and layout
    - JavaScript for interactivity and dynamic behavior
- **Back-End:**

    - Python 3 for server-side scripting
    - Django web framework for rapid development and efficient database interaction
- **Database:**
    - PostgreSQL or MySQL relational database management system for storing application data
- **API:**
    - OpenAPI framework for defining and documenting RESTful APIs

1.2 System Architecture

The system will follow a three-tier architecture:

- **Presentation Layer:** The user interface (UI) built with HTML, CSS, and JavaScript.
- **Business Logic Layer:** The back-end implemented with Django, handling user interactions, business logic, and communication with the database.
- **Data Layer:** The database (PostgreSQL or MySQL) storing application data.

1.3 Security Considerations

- **User Authentication and Authorization:** Implement a secure user authentication mechanism (e.g., username/password with hashing) and role-based authorization to restrict access to sensitive data and functionalities based on user roles.
- **Data Encryption:** Consider encrypting sensitive data (e.g., student roll numbers) at rest and in transit.
- **Session Management:** Implement secure session management practices to prevent unauthorized access(like outside of our organization).
- **Regular Security Updates:** Maintain the system and libraries used up-to-date with the latest security patches.

1.4 Performance Optimization

- **Database Indexing:** Properly index database tables to improve query performance.
- **Caching:** Implement caching mechanisms to store frequently accessed data in memory for faster retrieval.

- **Code Optimization:** Write efficient and optimized code to minimize server load and response times.

1.5 Scalability

- The system should be designed to scale horizontally by adding more server instances to handle increasing user load and data volume.
- Utilize cloud-based deployment options (if applicable) to leverage automatic scaling capabilities.

1.6 Testing

- Implement unit testing for back-end functionalities.
- Conduct integration testing to ensure different system components work together seamlessly.
- Perform user acceptance testing (UAT) with real users to validate system functionality and usability.

2. User Interface Mockups

A separate document can be created to showcase User Interface (UI) mockups for different functionalities of the External Events Management Portal. These mockups can be created using wireframing tools or design software (Figma).

2.1 Mockup

The UI mockups could include:

- Login screen
- Student dashboard
- Faculty dashboard
- Head dashboard
- Admin dashboard
- Event proposal submission form

- Event details page
- IRA registration page
- Slot allotment notification
- Reports dashboard (optional)

FIGMA : PROTOTYPE