SOFTWARE ENGINEERING LAB TASK 7

05-02-2025

HU22CSEN0100999

Eshwar Deshmukh Chavan

Create a SRS document on gathering requirements for University and student activities

1. Introduction

- **Purpose:** Define software requirements for managing student activities and events, ensuring efficient handling of academic and extracurricular engagements.
- **Scope:** Manage student registrations, event scheduling, activity tracking, attendance management, and provide administrative control for university staff.
- **Users:** Admin, Faculty, Students, Event Coordinators.

2. System Overview

- **Functions:**
    - Event management: Create, update, delete events.
    - Student registration for activities and events.
    - Attendance tracking and validation.
    - Reporting and analytics for event participation.
    - Notifications and reminders for upcoming events.
- **Users:**
    - Admin: Full system access to manage events, users, and reports.

    - Faculty: Schedule academic events, manage student participation.
    - Students: View, register, and track events.
    - Event Coordinators: Oversee event execution and attendance.
- **Platform:** Web-based, compatible with mobile & desktop devices, accessible via all major browsers.

3. Requirements

- **Functional Requirements:**
    - Create, edit, delete, and publish events.
    - Register and deregister for events.
    - Attendance tracking through QR codes or manual entry.
    - Generate detailed participation and attendance reports.
    - Role-based access for different user types.
    - Search and filter events by category, date, or organizer.
- **Non-Functional Requirements:**
    - Performance: Fast load times (2 seconds maximum).
    - Scalability: Handle 10,000+ concurrent users.
    - Security: Encrypted data, role-based access control, regular security audits.
    - Usability: User-friendly interface, intuitive navigation.
    - Reliability: 99.9% uptime with robust error handling mechanisms.

4. System Models

- **Use Case Diagrams:** For event creation, student registration, and report generation.

- **Sequence Diagrams:** Detailing user login/authentication, event registration process.
- **Data Flow Diagrams (DFD):** Illustrating data flow between modules like registration, events, and reporting.
- **Entity-Relationship Diagram (ERD):** Defining relationships between users, events, attendance, and reports.

5. Constraints

- Compliance with university IT security policies.
- Support for multiple operating systems (Windows, macOS, Linux).
- Adherence to web accessibility standards (WCAG 2.1).
- Limited to university network for administrative functions.
- Integration with existing university systems (e.g., Student Information System).

6. Assumptions

- Users have basic proficiency in using web applications.
- Reliable internet connectivity is available for all users.
- University IT department will provide regular system updates and maintenance.
- Events are pre-approved by the administration before publishing.

7. Dependencies

- Integration with university authentication system (SSO).
- Dependence on third-party libraries for specific functionalities like QR code generation.
- Hosting environment supported by university IT infrastructure.

8. Glossary

- **Admin:** System manager with full control.

- **Faculty:** Academic staff scheduling events.
- **Student:** End-user registering for activities.
- **Event Coordinator:** Person managing specific events.

9. Benefits

- **Improved Efficiency:** Automates event management and reduces administrative workload.
- **Enhanced Participation:** Simplified registration process encourages student involvement.
- **Data-Driven Decisions:** Detailed reports and analytics support better planning.
- **Accessibility:** Provides easy access from multiple devices and platforms.
- **Security:** Ensures data protection with encrypted communication and role-based access.
- **Scalability:** Can handle growing numbers of users and events without performance issues.