**Title: Event Management System**

**Submitted to: Dr Ghulam Mujtaba**

**Submitted by: Aaliyan Shujah**

**Reg No:- FA22-BCE-003**

**Date: 29 april,2025**

**1. Introduction**

* 1.1 Purpose

This document defines the Software Requirements Specification for the Event Management System. The system aims to simplify the process of organizing, registering, and managing events through an online platform. It is designed to support different types of events such as seminars, conferences, workshops, college fests, and cultural programs. The system is useful for both event organizers and attendees. Organizers can create events, manage registrations, track attendance, and communicate with participants. Participants can register for events, receive updates, and provide feedback after attending.

* 1.2 Scope

The Event Management System is a web-based application that provides a centralized platform for creating, promoting, and managing events. The main features of the system include event creation, online registration, user dashboards, attendance tracking, notifications, and reporting tools. It is intended to help event organizers reduce manual efforts and to make event information easily accessible to users. By using this system, organizations can improve event planning and enhance the participant experience.

* 1.3 Definitions, Acronyms, and Abbreviations

User – A person using the system, either as an event organizer or participant

Admin – The person responsible for managing users and system data

Registration – The process of signing up for an event

SRS – Software Requirements Specification

UI – User Interface

* 1.4 References

IEEE Software Requirements Specification guidelines

Documentation of similar online event platforms

University or organizational event management systems

**2. Overall Description**

* 2.1 Product Perspective

The Event Management System is a new standalone web application. It is designed to manage all aspects of event handling, from creation to reporting. The system will function independently but may be integrated with existing user databases or email services if needed. It is structured to allow multiple users to interact at the same time without performance issues.

* 2.2 User Needs and Product Functions

The system is designed to meet the needs of two major user groups: organizers and participants.

Organizers require the ability to:

- Create and update event details

- Manage event schedules and venues

- View and manage user registrations

- Communicate with registered participants

- Track attendance and generate reports

Participants expect to:

- Register easily for events

- View detailed event schedules

- Get notified about changes or updates

- Receive confirmation and reminders

- Submit feedback after attending events

* 2.3 Assumptions and Dependencies

Users must have access to a device with an internet connection

The system depends on a working web server and database server

Notification services may rely on third-party APIs such as email or SMS providers

**3. Specific Requirements**

**Functional Requirements**

* 3.1.1 User Registration and Login

Users can create an account by providing personal details such as name, email, and password. They can log in securely and manage their profiles. The system supports password reset and secure authentication.

* 3.1.2 Event Creation and Management

Organizers can create events by providing information like event title, date, time, venue, description, speaker details, and available seats. They can update or cancel the event at any time. The system should allow event categorization such as technical, cultural, academic, and entertainment.

* 3.1.3 Event Browsing and Search

Users can view upcoming events, use filters like date, category, or location, and see event details before registering. Each event page will include full information and a registration link.

* 3.1.4 Registration and Confirmation

Participants can register for events through an online form. Upon registration, the system sends confirmation via email or SMS. Organizers can view and manage the list of registered users.

* 3.1.5 Attendance Management

Organizers can mark attendance for participants either manually or using QR codes. The system records attendance and helps in generating participation certificates or reports.

* 3.1.6 Notifications and Messaging

Automatic notifications are sent for registration confirmation, reminders, updates, and cancellations. Organizers can also send messages to participants directly through the platform.

* 3.1.7 Feedback and Reporting

After the event, participants can submit feedback. The system generates reports for organizers including attendance, feedback summary, and event performance.

**3.2 Non-functional Requirements**

* 3.2.1 Usability

The system must be simple to use and suitable for users with basic computer knowledge. The design should be clean and mobile-friendly.

* 3.2.2 Performance

The system should load pages quickly and handle a large number of users and events without slowdowns.

* 3.2.3 Security

All user data should be protected using encryption. Communication between users and the system must be done over secure channels.

* 3.2.4 Availability and Reliability

The system should be available most of the time with minimal downtime. Backup procedures must be in place to restore data in case of failure.

**3.3 External Interface Requirements**

* 3.3.1 User Interface

The website should include clear menus, a homepage with event highlights, user dashboards for both organizers and participants, and easy navigation.

* 3.3.2 Hardware Interfaces

The application must run on standard web servers and be accessible from desktop and mobile devices.

* 3.3.3 Software Interfaces

The back-end should connect to a relational database for storing events and user data. It should support integration with email and SMS APIs.

* 3.3.4 Communication Interfaces

The system must use HTTP or HTTPS protocols. REST APIs may be used for integration with other systems or mobile applications.

* 3.4 Design Constraints

The application will be developed using standard web technologies such as HTML, CSS, JavaScript, and a server-side language like PHP or Python. It must be compatible with major browsers and should be tested thoroughly for performance and security.

* 3.5 Assumptions and Dependencies

The system assumes users will provide correct personal details when registering. It also depends on third-party services for delivering email or SMS notifications.

**4. System Architecture and Design Considerations**

* 4.1 Overall Architecture

The system follows a three-layer architecture:

Presentation layer – the web interface

Application layer – handles business logic

Data layer – manages the database

* 4.2 Data Flow and Process

The system captures data from user input, processes it based on the selected function (such as registration or feedback), stores it in the database, and provides output such as confirmation messages or reports.

* 4.3 User Interface Design

Mockups and wireframes should be created to show how users will interact with the system, including event creation forms, registration pages, and dashboards.

**5. Implementation Plan**

* 5.1 Development Environment

Front-end: HTML, CSS, JavaScript

Back-end: PHP, Python, or Node.js

Database: MySQL or PostgreSQL

Version Control: Git

Hosting: Cloud-based server or university server

**5.2 Timeline**

Phase 1: Requirement analysis and planning

Phase 2: UI/UX design and database setup

Phase 3: Core functionality development

Phase 4: Testing and bug fixing

Phase 5: Deployment and documentation

* 5.3 Testing Strategy

Unit testing for each module

Integration testing for combined components

User acceptance testing with feedback collection

Security and performance testing

**6. Maintenance and Future Enhancements**

* 6.1 Maintenance Plan

Regular updates to fix bugs and improve performance

Database backups and server monitoring

User support for login issues and event errors

* 6.2 Future Enhancements

Mobile app development

Integration with calendar services

Automated certificate generation

Support for hybrid or online events

Advanced analytics and event suggestions

**7. Appendices**

* 7.1 Glossary

Organizer – User who creates and manages events

Participant – User who registers and attends events

Attendance – Marking whether a user attended the event

Feedback – Comments submitted by users after the event

* 7.2 Supporting Materials

Wireframes, database schema, flowcharts, and UI mockups

* 7.3 Change Log

A record of all changes made to this document during development

**8. Summary and Approval**

This document defines the detailed requirements for the Event Management System. It provides a clear understanding of how the system will function, what features it will offer, and how users will interact with it. Once approved, it will serve as the foundation for development and implementation.