Software Requirements Specification

For

REVELS MANAGEMENT SYSTEM

Version 1.0

Prepared by

Akhil Varanasi 57 - 230953496

Saurabh Sharma 43 - 230953374

Shreyas Kumar PM 33- 230953300

Manipal Institute of Technology

1st April 2025

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this SRS document is to define, in comprehensive detail, the requirements for the **Fest Management System**. This system is designed to automate and manage various aspects of college fests including participant registration, event scheduling, judging, and team management. It aims to replace manual processes with a centralized, web-based solution that enhances efficiency, accuracy, and communication among all stakeholders.

1.2 Document Conventions

The formatting conventions for this document are as follows:

- **Font:** Times New Roman is used for all text. Regular text is size 12, while headings are in size 14 bold.
- **Emphasis:** Italics are used for emphasis, and bold is used for subpoints and key terms.
- **Layout:** The document is fully justified, ensuring a professional and uniform appearance.
- **Notation:** Each requirement is referenced using a unique identifier (e.g., FR-1 for functional requirements, NFR-1 for nonfunctional requirements).
- **Abbreviations:** Standard abbreviations (e.g., FMS for Fest Management System) are defined in Appendix A.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

- **Developers:** To design and implement the system according to the defined requirements.
- **Project Managers:** To oversee project progress and ensure alignment with the fest's objectives.
- **Event Organizers and Administrators:** To use the system for managing festival operations.
- **Judges:** To understand how to input scores and interact with the judging module.
- **Participants:** To register for events and track their progress.

- **Testers:** To validate that all functionalities meet the specified requirements.
- **Documentation Writers:** To create end-user manuals and training materials based on the SRS.

1.4 Product Scope

The **Fest Management System** is a comprehensive web-based application designed to simplify the management of college fests. Key features include:

- **Participant Registration:** Online forms for individuals and teams with validation and optional payment processing.
- **Event Scheduling:** Tools for creating, updating, and displaying a detailed event calendar with real-time notifications.
- **Judging System:** Secure interfaces for judges to input scores and provide feedback, with automated result calculation.
- **Team Management:** Capabilities for forming teams, assigning roles, and managing team registrations.
- **Result Calculation:** Algorithms to compile scores from multiple judges and display results dynamically.
- **Administrative Dashboard:** Centralized control for organizers to oversee all aspects of the fest.

1.5 References

- IEEE Std 830-1998 – Recommended Practice for Software Requirements Specifications.
- MySQL and PHP Official Documentation.
- Web Standards: HTML5, CSS3, and JavaScript guidelines.
- Best practices in user interface and experience design.

---

2. Overall Description

2.1 Product Perspective

The **Fest Management System** is an independent application designed to automate college fest operations. It integrates seamlessly with existing web technologies and third-party services (such as email servers and payment gateways) to deliver a cohesive experience. This system aims to reduce manual administrative tasks and improve real-time communication among all users.

2.2 Product Functions

Key functions of the system include:

- **User Authentication and Profile Management:** Secure registration, login, and profile maintenance for all user types.
- **Event Scheduling and Calendar Management:** Creation and management of event schedules with real-time updates.
- **Participant and Team Registration:** Simplified online registration process with support for both individual and team entries.
- **Judging and Scoring Module:** Secure interfaces for judges to submit scores and for the system to calculate final results automatically.
- **Notification System:** Automated email and SMS notifications for event updates, registration confirmations, and result announcements.

2.3 User Classes and Characteristics

- **Organizers/Administrators:**
    - *Characteristics:* Highly skilled in event planning and administration; require comprehensive control over the system.
    - *Responsibilities:* Create and manage events, oversee registrations, and maintain system configurations.

- **Participants:**
    - *Characteristics:* Students or individuals with limited technical skills; require a simple and intuitive interface for registration and event participation.
    - *Responsibilities:* Register for events, manage personal profiles, and receive notifications.
- **Judges:**
    - *Characteristics:* Subject matter experts or faculty members with a need for secure and reliable scoring mechanisms.
    - *Responsibilities:* Input scores, provide feedback, and review participant performance.
- **Guests/Audience:**
    - *Characteristics:* Individuals interested in viewing event schedules and results; require read-only access.
    - *Responsibilities:* Access public event information and results.

2.4 Operating Environment

- **Server Environment:**
    - Web servers running Apache or Nginx with PHP and MySQL as the backend database.
- **Client Environment:**
    - Accessible on desktops, laptops, tablets, and smartphones via modern web browsers such as Chrome, Firefox, Safari, and Edge.
- **Operating Systems:**
    - Compatible with Windows, macOS, Linux, iOS, and Android.

2.5 Design and Implementation Constraints

- **Scalability:** Must support a high volume of concurrent users during peak registration times.
- **Security:** Requires robust data encryption, secure authentication, and regular vulnerability assessments.
- **Compatibility:** Designed to work on multiple platforms and browsers without performance degradation.
- **Maintainability:** Code should be modular and thoroughly documented to facilitate future updates and feature enhancements.

2.6 User Documentation

User documentation will be provided to ensure smooth adoption and operation of the system. It will include:

- **User Manuals:** Detailed guides for organizers, participants, and judges outlining system usage, troubleshooting steps, and FAQs.
- **Online Help:** A dedicated help portal with searchable knowledge bases and interactive support forums.
- **Tutorial Videos:** Step-by-step video tutorials covering key functionalities and workflows.

2.7 Assumptions and Dependencies

- **Assumptions:**
    - All users have stable internet connectivity during festival periods.
    - Organizers, judges, and participants are familiar with basic web interfaces.
    - The institution will provide necessary support for server hosting and maintenance.
- **Dependencies:**
    - Reliance on third-party services such as SMTP servers for email notifications and, if applicable, external payment gateways.

- Future mobile app integration may require additional resources.

---

3. External Interface Requirements

3.1 User Interfaces

The system will provide multiple user interfaces tailored to each user class:

- **Login/Registration Interface:** A secure portal for new user registration and existing user authentication.
- **Dashboard:** Custom dashboards for organizers, participants, and judges that display relevant information such as event schedules, registration status, and scoring details.
- **Event Calendar:** A dynamic, interactive calendar view displaying all scheduled events with filtering options.
- **Judging Interface:** A simplified interface for judges to submit scores and comments, with data validation and secure access.
- **Team Management Interface:** Tools for creating and managing teams, with options for inviting members and assigning roles.

3.2 Hardware Interfaces

- **Server Hardware:**
  - Standard physical or cloud servers equipped with sufficient processing power, memory, and storage to handle peak loads.
- **Client Devices:**
  - Any device with a modern web browser including desktops, laptops, tablets, and smartphones.

3.3 Software Interfaces

- **Database:**
    - MySQL database for storing user data, event details, registration records, scores, and other system information.
- **Backend:**
    - PHP for server-side scripting and business logic implementation.
- **Frontend:**
    - HTML, CSS, and JavaScript for rendering interactive user interfaces and ensuring responsiveness.

3.4 Communications Interfaces

- **Web Protocols:**
    - HTTP/HTTPS for secure communication between client and server.
- **Email Communication:**
    - Integration with SMTP servers to send registration confirmations, notifications, and password reset emails.
- **APIs:**
    - RESTful APIs will be used to facilitate communication between the frontend and backend, as well as any external integrations (e.g., payment gateways).

---

4. System Features

4.1 Participant Registration

Description:
Provide a secure, user-friendly registration process allowing individuals or teams to sign up for events.

Functional Requirements:

- FR-1: Validate user input for registration forms (e.g., name, email, contact number, college).
- FR-2: Store registration data securely in the MySQL database.
- FR-3: Provide confirmation emails upon successful registration.
- FR-4: Support optional payment processing if event fees apply.

4.2 Event Scheduling

Description:
Enable organizers to create, update, and display event schedules.
Functional Requirements:

- FR-5: Allow organizers to add new events with details such as event name, date, time, and venue.
- FR-6: Provide an interactive calendar view for displaying scheduled events.
- FR-7: Automatically send notifications to registered users when events are updated or rescheduled.

4.3 Judging System

Description:
Offer a secure interface for judges to submit scores and comments for events.
Functional Requirements:

- FR-8: Authenticate judges through secure login procedures.
- FR-9: Allow judges to enter scores and qualitative feedback for each participant or team.
- FR-10: Validate score inputs to ensure they meet defined criteria and ranges.
- FR-11: Automatically calculate cumulative scores and update results in real time.

4.4 Team Management

Description:
Support the formation and management of teams for events that require group participation.
Functional Requirements:

- FR-12: Enable participants to create or join teams, with appropriate verification.
- FR-13: Allow team leaders to manage team member roles and update team information.
- FR-14: Ensure team data is linked to participant registrations and event details.

4.5 Result Calculation

Description:
Automate the process of compiling scores from multiple judges and generating final results.
Functional Requirements:

- FR-15: Aggregate scores from all judges for each event.
- FR-16: Apply pre-defined business rules to compute final rankings.
- FR-17: Publish results on user dashboards and public result pages with options for detailed score breakdowns.

---

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- NFR-1: The system shall support at least 1000 concurrent users without performance degradation.
- NFR-2: Average page load times must be less than 3 seconds under normal operating conditions.
- NFR-3: The system should process registration and scoring inputs in real time.

5.2 Safety Requirements

- NFR-4: Regular data backups must be performed to prevent loss of information.
- NFR-5: A disaster recovery plan must be in place to ensure minimal downtime in the event of system failure.

5.3 Security Requirements

- NFR-6: The system must implement role-based access control, ensuring that users only have access to appropriate modules.
- NFR-7: All data transmitted between the client and server must be encrypted using HTTPS.
- NFR-8: Sensitive data (e.g., passwords, payment details) must be stored using secure, industry-standard encryption methods.

5.4 Software Quality Attributes

- **Usability:**
    - NFR-9: The user interface must be intuitive, with clear navigation and consistent design patterns.
- **Maintainability:**
    - NFR-10: The codebase must be modular and well-documented to facilitate future updates and maintenance.
- **Scalability:**
    - NFR-11: The system architecture should support easy scaling to accommodate increased user load.
- **Reliability:**
    - NFR-12: The system should achieve at least 99.5% uptime, with robust error handling and logging mechanisms.

5.5 Business Rules

- NFR-13: Only registered users may participate in events or join teams.
- NFR-14: Judges must verify participant eligibility before submitting scores.
- NFR-15: Changes to event schedules must be approved by an organizer and trigger automatic notifications.
- NFR-16: Financial transactions, if applicable, must be logged for audit purposes and comply with institutional policies.

---

6. Other Requirements

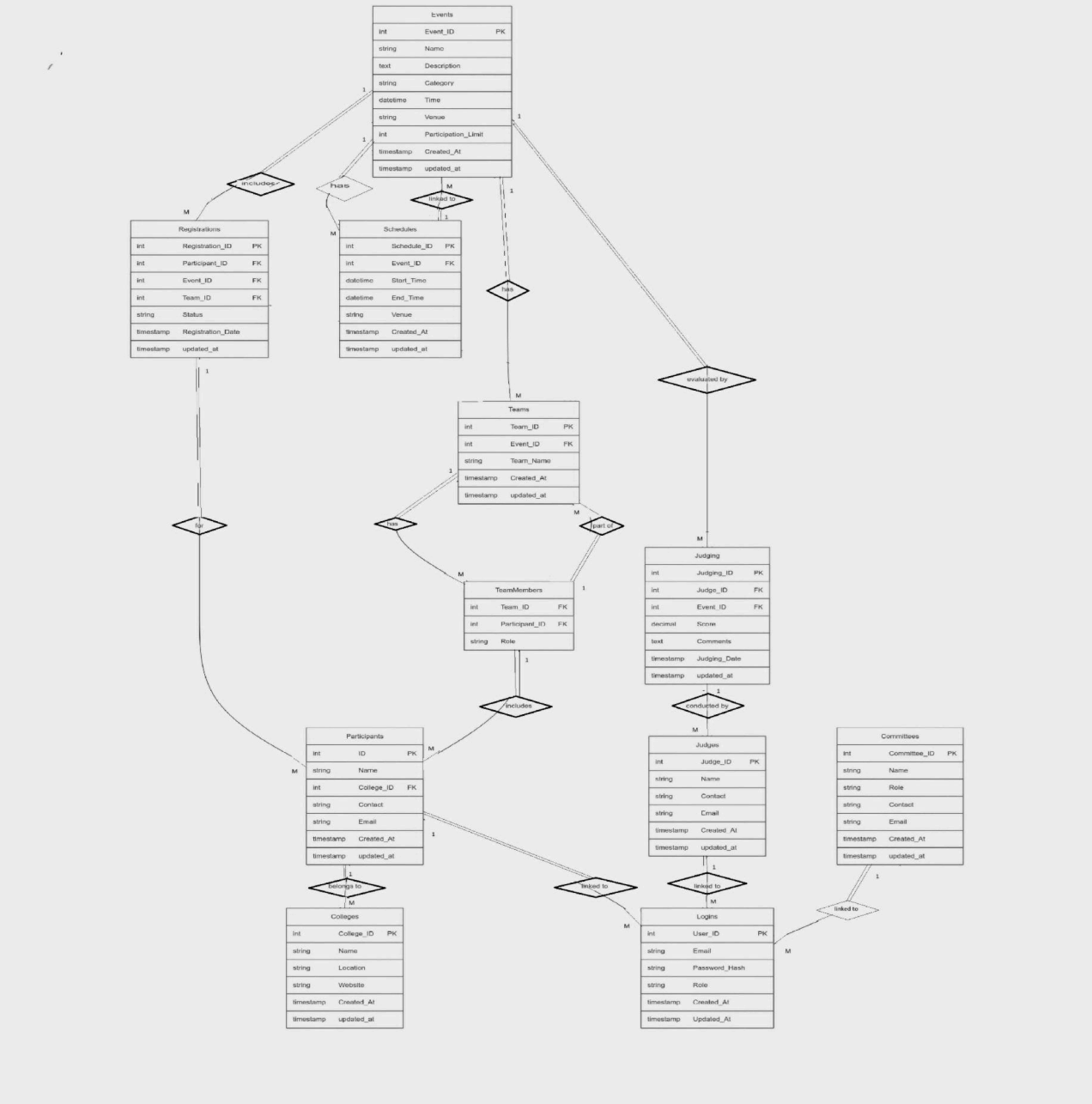
- **Database Requirements:**
    - The system will utilize a MySQL database designed to handle large volumes of data and complex queries efficiently.
- **Internationalization:**
    - While initially deployed for a single locale, the system should be designed to support multiple languages and regional settings in future releases.
- **Legal Requirements:**
    - The system must comply with data protection regulations (e.g., GDPR) and institutional policies regarding data privacy and security.
- **Reuse Objectives:**
    - Core modules such as user management and event scheduling should be designed for potential reuse in other related projects.

Appendix A: Glossary

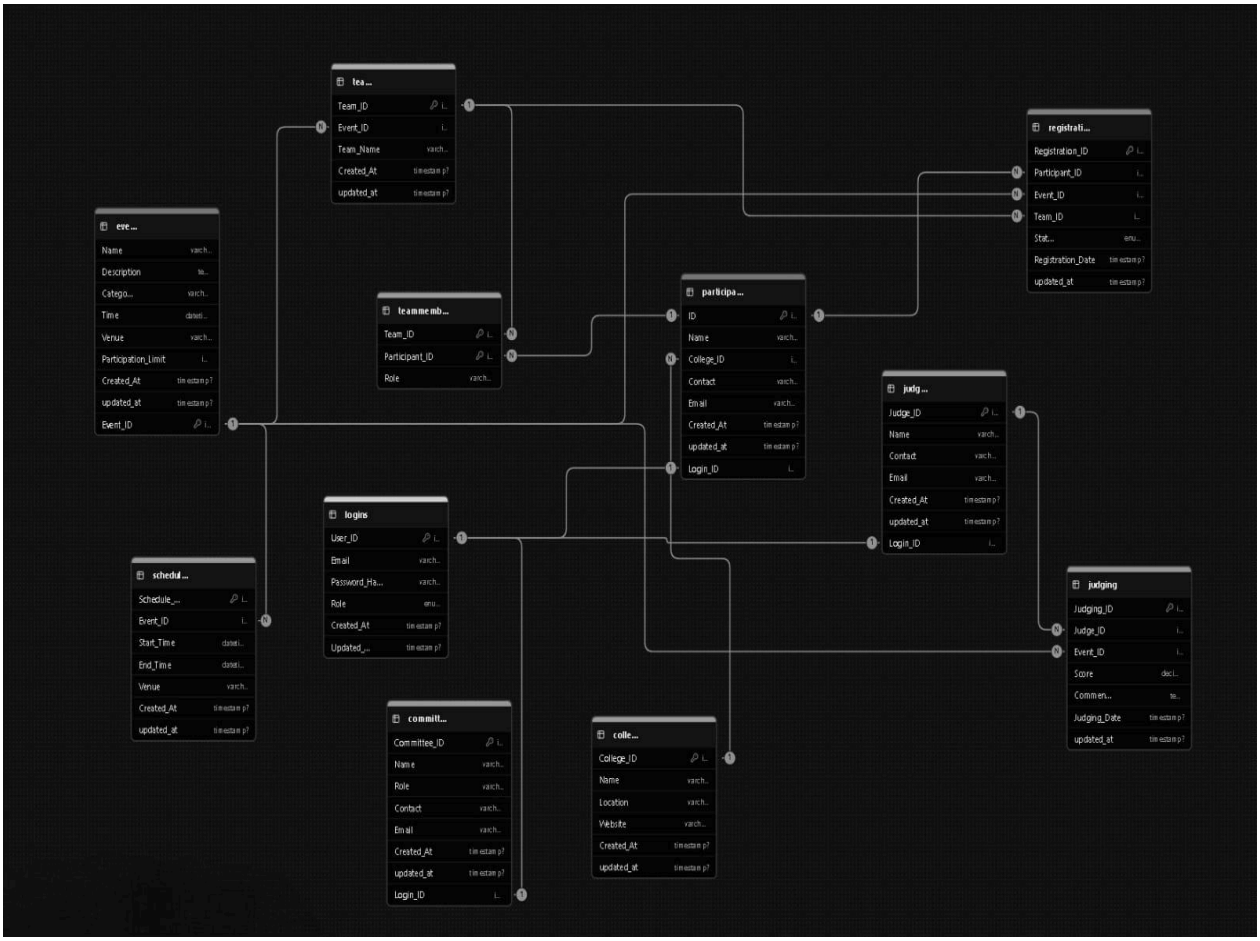
- **None**

Appendix B: Analysis Models

- **Entity-Relationship Diagrams (ERD):** Diagrams depicting the relationships between key data entities such as Users, Events, Teams, and Scores.



- **Schema Diagram:** A visual representation of a database structure. It shows the tables, columns, data types, primary keys, foreign keys, and relationships between tables in a database.



Appendix C: To Be Determined List

- None