**SOFTWARE REQUIREMENTS SPECIFICATION**

**For**

**Event Management System**

**Prepared by :-**

**TEAM 2**

1. Karthick M G

2. Sowbharnika P G

3. Sugirtha Vardhini M

**1. Introduction:**

The primary purpose of the Event Management System is to automate and simplify the process of event planning, coordination, and execution. It aims to centralize all aspects of event management, including event creation, attendee registration, resource allocation, scheduling, and reporting. By utilizing this system, organizers can save time, reduce errors, and enhance the overall experience for both organizers and attendees. The Event Management System (EMS) is a comprehensive software solution designed to streamline the planning, organization, and execution of events of varying scales and complexities. Developed using Java, this system aims to provide a user-friendly interface and robust functionality to facilitate efficient management of events for organizers and participants alike.

**1.1 Scope :**

The Event Management System encompasses a wide range of functionalities, including:

Event Creation: Users can create different types of events, specifying details such as date, time, location, and event type.

Registration and Ticketing: Attendees can register for events, purchase tickets, and receive confirmations via the system.

Resource Management: Efficient allocation and management of resources such as venues, equipment, and personnel required for events.

Scheduling and Calendar Integration: A calendar feature to schedule events, avoid conflicts, and integrate with external calendars.

Reporting and Analytics: Comprehensive reporting tools to track attendance, revenue, feedback, and other key metrics for event analysis.

**1.2 Objectives**

The objectives of the Event Management System project are to: Develop a user-friendly interface accessible to both organizers and attendees. Implement secure user authentication and authorization mechanisms. Ensure scalability and flexibility to accommodate various types of events and user requirements. Enable seamless communication between organizers and participants. Provide robust data management and reporting capabilities for effective decision-making.

This Software Requirements Specification (SRS) document outlines the functional and non-functional requirements, system architecture, user interfaces, and other crucial aspects of the Event Management System.

**1.3 Definitions, Acronyms and Abbreviations :**

JAVA -> platform independence

SQL-> Structured query Language

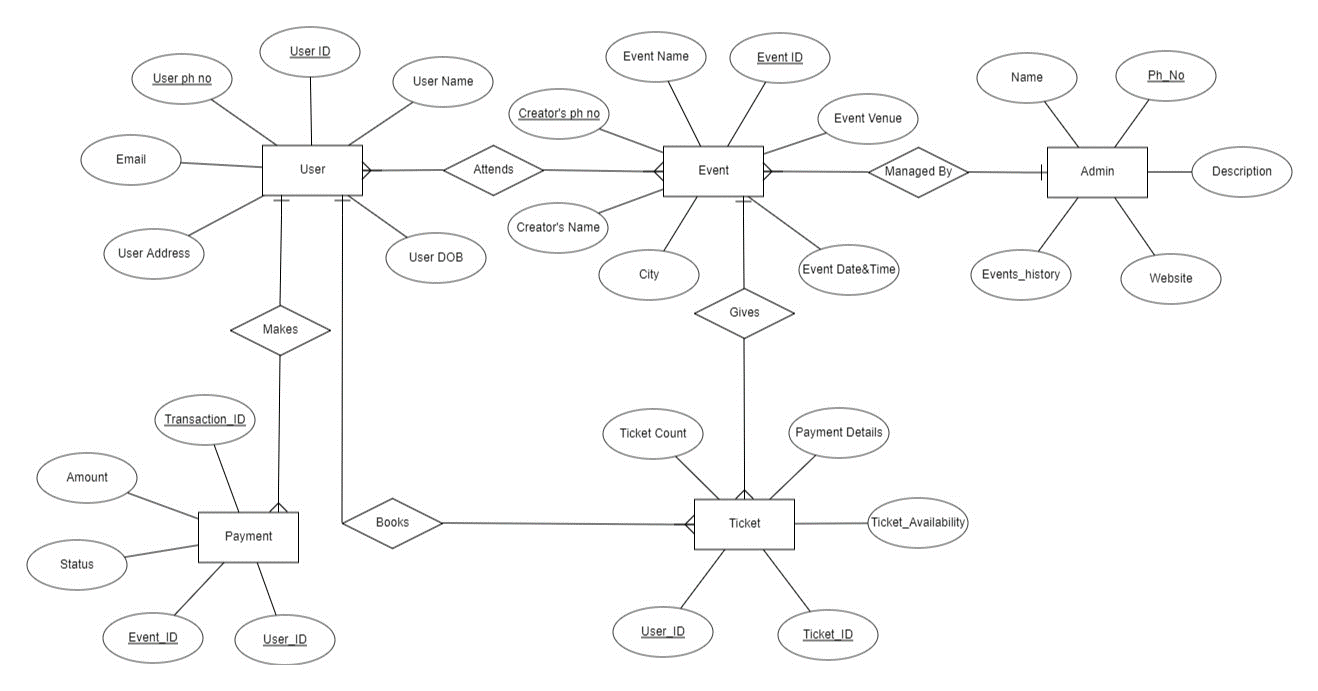
ER-> Entity Relationship

UML -> Unified Modeling Language

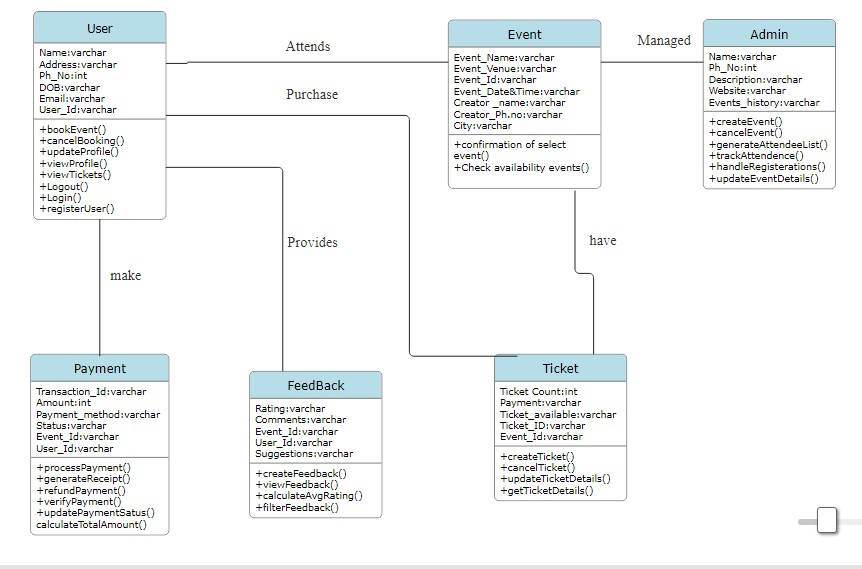
IDE-> Integrated Development Environment

SRS-> Software Requirement Specification

**1.4 ER Diagram:**



**1.5 Class Diagram:**



**2. Product Features :**

This program designed to assist managers, event organizers, firms, and other users whose line of business deals with events management to manage their participants’ data in an orderly manner. It shall perform the following functions:

• Protect the database of the firm by requiring a correct and registered username and password.

• Facilitate a step-by-step process of entering, organizing, retrieving, modifying and deleting data from the database without the need to go the database itself.

• Add new client information easily.

• Provide an option for users to update information.

• Delete existing client information.

• Display client information in an organized manner for easy understandability.

**3. Software Requirements**

Java Development Kit (JDK) 11

JavaFX SDK 11

Database: MySQL or PostgreSQL

IDE: IntelliJ IDEA or Eclipse

**4. System Features :**

**Event Creation and Management :**

Create new events with details such as date, time, location, and type.

Edit and update event information.

Set event capacity and limits.

**Registration and Ticketing:**

User registration and login for attendees, organizers, and administrators.

Purchase and issuance of tickets.

Ticket scanning/verification at the event.

**Venue Management:**

Availability and booking of venues.

Managing multiple venues and their capacities.

Visual floor plans or seating arrangements.

**Resource Allocation:**

Equipment and resource allocation for different events.

Vendor management for external resource booking.

**Scheduling and Calendar Integration:**

Schedule events without conflicts.

Integration with external calendars (Google Calendar, Outlook, etc.).

**Communication and Notifications:**

Automated event reminders and notifications to attendees.

Internal communication platform for organizers, attendees, and vendors.

**Payment and Invoicing:**

Online payment processing and secure transactions.

Invoicing and financial tracking for event budgets.

**Feedback and Surveys:**

Collect post-event feedback from attendees.

Surveys to gather suggestions for improvements.

**Reporting and Analytics:**

Generate reports on attendance, revenue, feedback, etc.

Data analytics for insights and decision-making.

**Customization and Branding:**

Customizable event pages and branding options for organizers.

Personalized experiences for attendees.

**Social Media Integration:**

Share events on social media platforms.

Integration with social media for event promotion and engagement.

**5. Security Requirements :**

Ensuring safety in an event management system is crucial, and it involves considering various aspects to protect both users and data. Here are some safety requirements you might want to consider for an event management system developed in Java:

**User Authentication and Authorization:**

Implement secure user authentication mechanisms, such as password hashing and salting.Use role-based access control (RBAC) to ensure that users have appropriate permissions.Regularly update and review user access levels to prevent unauthorized access.

**Data Encryption:**

Use encryption for sensitive data, especially during transmission (HTTPS) and storage.Employ strong encryption algorithms for password storage.

**Input Validation:**

Implement strict input validation to prevent SQL injection, cross-site scripting (XSS), and other injection attacks. Validate and sanitize user inputs on both client and server sides.

**Secure File Uploads:**

If the system involves file uploads, implement strict controls and validation to prevent malicious file uploads. Set restrictions on file types, sizes, and enforce proper file handling practices.

**Secure Communication:**

Use secure communication protocols (TLS/SSL) to protect data during transit. Avoid hardcoding sensitive information such as API keys, passwords, or encryption keys in the source code.

**Data Backups:**

Regularly backup critical data to prevent data loss in the event of system failures or security incidents. Ensure that backup data is encrypted and stored in a secure location.

**6. Safety Requirements:**

Different information is entered into the database such as information about the different caterers, suppliers and participants. Mismanagement of information might cause participant dissatisfaction that will eventually lead to profit loss, only because of mistakes on giving information. In line with this, the organizer should always double check which suppliers are available.

**References:**

<https://www.scribd.com/document/167542681/project-report-on-event-management-pdf>

<https://www.freeprojectz.com/java-jsp-mysql-project-download/event-management-system>

<https://www.codewithc.com/event-management-system-project-java/>