Software Requirements Specification

for

<Event Management System>

Version 1.0 approved.

Prepared by <RE-2023-002>

<15/12/2023>

Table of Contents

1. Introduction

1.1 Purpose

This document, known as the Software Requirements Specification (SRS), is dedicated to defining the specifics for the Event Management System (EMS), specifically Version 1.0. It comprehensively covers all aspects of the EMS, including event creation, attendee registration, payment handling, and participant management. The EMS is meticulously crafted to simplify tasks for event organizers, attendees, and sponsors, facilitating the seamless organization of various events, ranging from small gatherings to large conferences.

This document serves as a guide, explaining the capabilities of the software and how it functions. It also functions as a reference for future updates or improvements. The System Overview section elucidates that the EMS is designed to ensure the seamless execution of events for organizers, enhanced engagement for attendees, and streamlined collaboration for sponsors. The Revision History keeps track of any changes made to the document, starting with the initial version labeled as 1.0.

1.2 Document Conventions

This Software Requirements Specification (SRS) adheres to specific standards and typographical conventions to ensure clarity and consistency throughout the document.

1.2.1 Typography

- Headings and subheadings: Times, 14-point bold
- Body text: Times New Roman, 12-point
- Code or technical terms: Courier New, 10-point

1.2.2 Highlighting

- Requirements: Bold text
- Important notes or warnings: Italicized text

1.2.3 Priority

Each requirement statement is assigned its priority level to indicate its significance.
Priorities are denoted as follows:

- Priority 1: Critical, must-have
- Priority 2: Important, but not critical
- Priority 3: Nice-to-have, can be considered for future enhancements

1.2.4 Inheritance of Priorities

Priorities for higher-level requirements are assumed to be inherited by detailed requirements unless explicitly stated otherwise.

These conventions are employed to enhance the document's readability and provide a standardized framework for understanding requirements and their respective priorities.

1.3 Intended Audience and Reading Suggestions

This Software Requirements Specification (SRS) is crafted to address a diverse audience involved in the development, implementation, and evaluation of the Event Management System (EMS). The document anticipates the following categories of readers:

1) Developers:
    - Developers will find detailed technical specifications for implementing the EMS. Relevant sections include "System Architecture" and "Functional Requirements."

2) Project Managers:
    - Project managers can gain insights into the project scope, priorities, and timelines by referring to the "Scope" and "Project Timeline" sections.

3) Users:
    - End users, including event organizers, sponsors and participants, will benefit from the "User Interface Requirements" section, offering a glimpse into the system's user experience.

4) Testers:
    - Testers should focus on the "Functional Requirements" and "Non-functional Requirements" sections to derive test cases and evaluate system performance.

5) Documentation Writers:
    - Those responsible for creating user manuals or helping with documentation can refer to the "System Overview" and "User Interface Requirements" sections for comprehensive understanding.

Document Organization:

- Section 1 (Introduction): Provides an overview of the document's purpose and scope.
- Section 1.1 (Identification): Specifies the product and version details.
- Section 1.2 (Document Conventions): Describes the standards and conventions used in the document.
- Section 1.3 (Intended Audience and Reading Suggestions): Defines the audience and suggests reading sequences.
- Remaining Sections: Organized for specific reader types.
    - ✓ Developers: "System Architecture," "Functional Requirements."
    - ✓ Project Managers: "Scope," "Project Timeline."
    - ✓ Users: "User Interface Requirements."
    - ✓ Testers: "Functional Requirements," "Non-functional Requirements."

- ✓ Documentation Writers: "System Overview," "User Interface Requirements."

This structure is designed to facilitate efficient comprehension for each reader type, streamlining the extraction of pertinent information.

1.4 Project Scope

The Event Management System (EMS) is a comprehensive software solution designed to facilitate the planning, organization, and management of diverse events. The primary purpose of the EMS is to streamline and enhance the operational efficiency of both event organizers and participants.

Description:

The EMS encompasses key functionalities, including but not limited to event creation, attendee registration, payment processing, and participant management. Event organizers can leverage the system to seamlessly coordinate various aspects of event planning, while participants benefit from an intuitive and user-friendly platform for event registration and engagement.

Objectives and Goals:

- Simplify Event Planning: The EMS aims to simplify the often-complex process of planning and organizing events, providing organizers with user-friendly tools and interfaces.
- Enhance Participant Experience: Participants can easily discover, register for, and engage in events through a user interface designed for simplicity and efficiency.
- Improve Operational Efficiency: By automating tasks such as attendee registration and payment processing, the EMS enhances overall operational efficiency for event organizers.

Corporate Goals and Business Strategies:

The development of the EMS aligns with broader corporate goals and business strategies by:

- Meeting Market Demand: Addressing the growing demand for effective event management solutions in a variety of sectors.
- Enhancing Brand Reputation: Providing a reliable and feature-rich EMS contributes to the positive perception of our brand in the market.
- Generating Revenue: Offering the EMS as a service creates revenue opportunities, particularly in sectors heavily reliant on successful event execution.

Relationship to Long-Term Vision:

This SRS serves as a subset of the long-term strategic vision for the EMS. As the system evolves, future releases will continue to align with the overarching goal of providing a leading-edge event management solution that adapts to emerging industry trends and user expectations.
In the absence of a separate vision and scope document, this section provides a concise overview of the EMS's purpose, benefits, objectives, and alignment with corporate goals.

1.5 References

1. ER Diagram representation. (n.d.). Tutorialspoint. Retrieved from https://www.tutorialspoint.com/dbms/er_diagram_representation.htm
2. Bandakkanavar, R. (2023, May 8). Software Requirements Specification document with example. Krazytech. Retrieved from https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database
3. International Institute of Business Analysis (IIBA). (2009). A Guide to the Business Analysis Body of Knowledge (BABOK Guide) Version 2.0. Ontario: IIBA.
4. Software Engineering Data Flow Diagrams - javatpoint. (n.d.). Javatpoint. Retrieved from https://www.javatpoint.com/software-engineering-data-flow-diagrams

2. Overall Description

2.1 Product Perspective

Event management software is a standalone product developed to streamline and enhance the management of events. It is not part of a larger system but is designed to integrate seamlessly with other software and services. The software is intended to be a comprehensive solution for event organizers, providing tools for event registration, ticketing, session and agenda management, sponsor management, and payment and financial management.

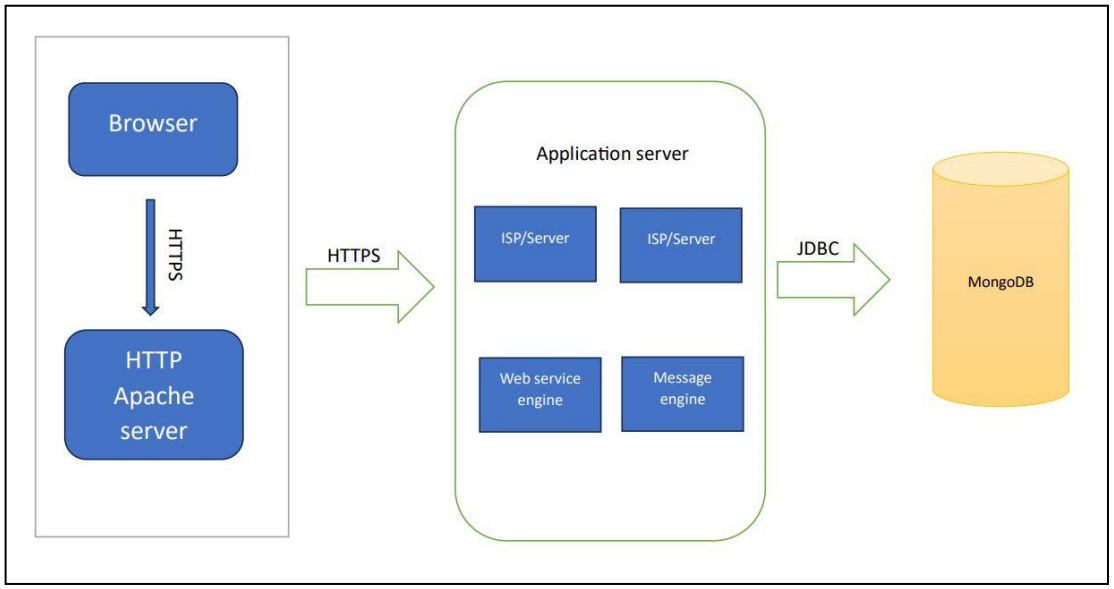

Figure 1: High level Architecture

2.2 Product Features

Our product general functions are:

1. Event Registration and Ticketing:

- Event Organizers:
  - ✓ Create and customize registration forms.
  - ✓ Set up different ticket types (e.g., early bird, VIP) with varying prices.
  - ✓ Send automated confirmation emails to registrants.

- Registration Team:
  - ✓ Manage attendee lists and registrations.
  - ✓ Assist with on-site check-in using the software.
  - ✓ Handle inquiries and provide support for registration-related issues.
  - ✓ Generate and analyze registration reports.

- Attendees:
  - ✓ Easily register for events online.
  - ✓ Access and update personal information.
  - ✓ Receive electronic tickets and event confirmations.
  - ✓ View and manage their event itinerary.

2. Session and Agenda Management:

- Event Organizers:
  - ✓ Create and manage event schedules and agendas.
  - ✓ Assign speakers and presenters to specific sessions.
  - ✓ Allow for easy updates to the agenda in real time.
  - ✓ Collect and manage presentation materials.

- Attendees:
  - ✓ View detailed event schedules and agendas.
  - ✓ Create a personalized agenda or itinerary.
  - ✓ Receive notifications for upcoming sessions.
  - ✓ Access presentation materials post-event.

3. Sponsor Management:

- Event Organizers:
  - ✓ Manage sponsor profiles and contact information.
  - ✓ Offer various sponsorship packages and benefits.
  - ✓ Track sponsor commitments and deliverables.
  - ✓ Provide visibility for sponsors through event materials.

- Sponsors:
  - ✓ Access information on sponsorship packages and benefits.
  - ✓ Submit required materials and logos.
  - ✓ Monitor exposure and engagement metrics.

4. Payment and Financial Management:

- Event Organizers:
  - ✓ Integrate secure payment gateways for ticket sales.
  - ✓ Track financial transactions and revenue in real time.

- Financial Team:
  - ✓ Access detailed financial reports.
  - ✓ Track expenses against the budget.
  - ✓ Manage invoices and payments.
  - ✓ Ensure compliance with financial regulations.

- Registration Team:
  - ✓ Monitor payment status for registrations.
  - ✓ Resolve payment-related issues and discrepancies.

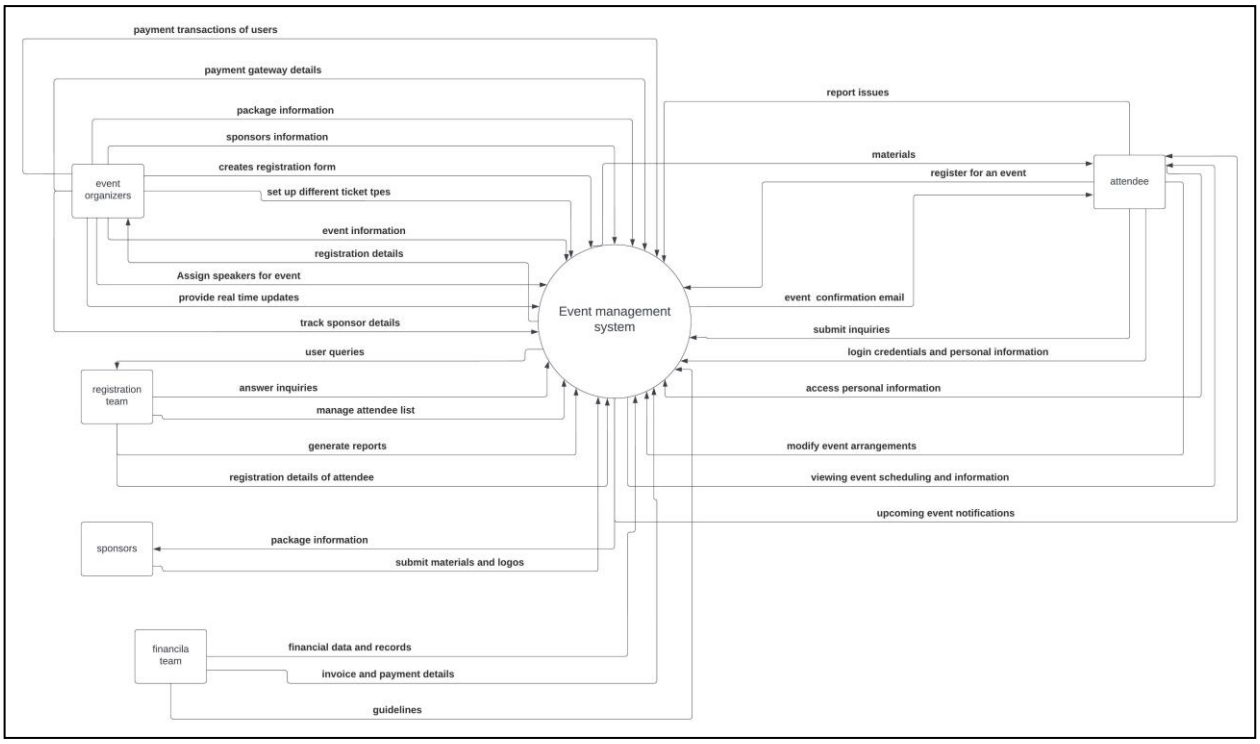

Figure 2: Data flow diagram

2.3 User Classes and Characteristics

User class	Frequency of Use	Technical Expertise	Functions	Access control level	Privileges	Educational level
Event Organizers	Weekly to daily	Experienced	Event planning, Execution, and management	Administrator Level	Full access, including event configuration and financial management	Varies (Professional)
Attendees	Monthly to weekly	Varies	Event registration, schedule access	User level	Limited access, focusing on personal event-related activities	Varies (Any)
Sponsors	Monthly to weekly	Moderate	Sponsorship management, profile handling, assessment	Sponsor level	Specific access to sponsorship-related features	Varies (Professional/ Business)
Financial team	Weekly to daily	Experienced	Financial management, Budgeting, Expense tracking	Administrator level	Full access, including financial configuration, budget allocation, and expense approval	Professional
Registration team	Weekly to daily	Moderate	Event registration, attendee management, ticket handling	User level	Access to registration-related features, attendee data management	Varies (Any)

2.4 Operating Environment

The software will operate in a diverse environment:

- Hardware: Compatible with standard hardware configurations.
- Operating System: Compatible with major operating systems (Windows, macOS, Linux).
- Software Components: Integrates with common web browsers, payment gateways, and third-party APIs.

2.5 Design and Implementation Constraints

The following constraints are identified:

- Budget: Development must adhere to a budget range of LKR 200,000 to LKR.300,000.
- Timeframe: The project has a six-month timeline for completion.
- User Load: The system should accommodate up to 1,500 concurrent users.
- Technology Stack: Specifies the use of certain technologies, tools, and databases.
- High security

2.6 User Documentation

The following documents will be delivered along with the software for assistance:

- User manuals (pdf)
- FAQs (Online FAQ page)
- Release notes (In- App notification)
- Interactive demos (downloadable demos)

2.7 Assumptions and Dependencies

Assumptions:

- Third-Party Components: Assumes the availability and functionality of third-party components.
- Stable Development Environment: Assumes a stable development and operating environment throughout the project.

Dependencies:

- External Services: Dependencies on external services, such as payment gateways and API providers.
- Regulatory Compliance: Dependency on adherence to corporate and regulatory policies.

3. System Features

3.1 Event Registration and Ticketing

i. Event Organizers

F1	Create and customize event registration forms
Input	Event details, designs, and styling preferences
Process	Design and generate a registration form based on the provided details.
Output	Customized registration forms
Definition	allows organizers to create unique registration forms by providing details on field names, types, and styling preferences. The system then generates a personalized registration form based on these inputs.

F2	Set up different ticket types
Input	Details about the types of tickets, including names, prices, and any specific attributes
Process	Ticket setup module, pricing configuration
Output	Various ticket types with varied prices and specifications
Definition	enables organizers to establish distinct ticket categories by providing details such as names, prices, and attributes. The system is then configured to recognize and manage these different ticket types accordingly.

F3	Automate confirmation emails
Input	Registration data, email templates
Process	Automated email system triggers upon registrations
Output	Automated confirmation emails sent to registrants
Definition	F3 automates the process of sending confirmation emails to attendees who successfully register for an event. The system takes in registration details, such as attendee information and event specifics, and immediately triggers confirmation emails containing relevant event details to provide a seamless and instant confirmation experience for the attendees.

ii. Registration Team

F4	Manage attendee lists and registrations
Input	Registration details of attendees
Process	Dashboard for managing registrations, attendee list management
Output	Well-organized attendee lists and up-to-date registration records
Definition	organizing registration details into lists, maintaining up-to-date records through a user-friendly interface. It streamlines the process, making it easy for users to organize and maintain accurate attendee information.

F5	Attendee Inquiry and Support Handling
Input	Registrants' queries, support requests
Process	Support ticket system, knowledge base/FAQ section
Output	Query resolution, support assistance
Definition	manages attendee inquiries and support requests. The system processes incoming queries, providing timely and helpful responses to ensure attendee satisfaction.

F6	Report generation and Analysis
Input	Registration data, reporting requirements
Process	Automatically generate reports based on registration data. Provide tools for analysis, summarization, and visualization of registration information.
Output	Comprehensive reports and insights derived from registration data
Definition	automates the creation of reports from registration data. Team can input parameters to tailor the reports to their needs. The system not only generates comprehensive reports but also provides tools for analysis, summarization, and visualization of registration information, offering valuable insights.

iii. Attendees

F7	Online event registration
Input	Attendee details, event preferences, and required information for registration
Process	User-friendly registration interface, form processing
Output	Quick and efficient event registration
Definition	Simple and accessible online event registration

F8	Access and update personal information
Input	User login credentials, personal details
Process	Enable users to securely access and update their personal information, implement authentication measures to ensure data privacy.
Output	Updated personal information, profile management
Definition	allows users to securely access and update their personal information. By providing user credentials, individuals can modify their details, and the system ensures data privacy through authentication measures. The output is updated and secure personal information records.

F9	Electronic Ticket Issuance
Input	Successful event registration details
Process	Ticket generation system, email delivery
Output	Electronic tickets with QR codes for event access
Definition	automatically generates electronic tickets for individuals who successfully register for an event. These electronic tickets are delivered to registrants either through email

F10	Event itinerary management
Input	Registered event data, itinerary management requests
Process	Personalized itinerary interface, schedule management
Output	Customized event itinerary, schedule updates
Definition	facilitates the management of the event itinerary. Attendees, upon authentication, can easily view and modify the event schedule using a user-friendly interface. The system allows for additions, modifications, or removals of events, ensuring that users have an updated and personalized event itinerary.

3.2 Session and Agenda Management

i. Event Organizers

F11	Event Schedule Creation and Management
Input	Details about event sessions, timings, and any specific scheduling preferences.
Process	Schedule creation tools, agenda management interface
Output	Organized event schedules and agendas
Definition	Creation and organization of event schedules and agendas

F112	Assign speakers and presenters
Input	Speaker details, session allocation
Process	Speaker assignment module, session-speaker mapping
Output	Designated speakers and presenters for each session
Definition	streamlines the process of assigning speakers and presenters to specific event sessions.

F13	Allow real-time updates to the agenda
Input	Update requests, agenda modification requests
Process	Real-time editing functionality, revision control
Output	Updated event agendas with real-time modifications
Definition	Flexibility for live updates and modifications to agendas

F14	Presentation Material Collection and Management
Input	Uploaded presentation files, documents, or multimedia materials.
Process	Presentation submission interface, material management tools
Output	Organized collection and management of presentation materials
Definition	Gathering and organizing session-related presentation materials

ii. Attendees

F15	Detailed Event Schedule Viewing
Input	Access to the event schedule through user credentials.
Process	Detailed schedule display, navigation tools
Output	Detailed and easily accessible event schedules.
Definition	enables attendees to view detailed event schedules through a user-friendly interface. With access granted through user credentials, users can easily navigate, search, and filter through the schedule details.

F16	Upcoming event notifications
Input	User preferences, session selections
Process	Automatically send notifications to users for upcoming events based on their preferences, include options for configuring notification frequency and content
Output	Timely alerts about upcoming sessions
Definition	ensures users receive timely notifications for upcoming events. Users can set their notification preferences, and the system automatically sends notifications based on those preferences

F17	Post-Event Presentation Material Access
Input	Access request, session details
Process	Post-event access portal, document retrieval
Output	Availability of session materials for post-event access
Definition	Retrieval and access to session-related presentation materials post-event

3.3 Sponsor Management

i. Event Organizers

F18	Sponsor profile and contact management
Input	Sponsor details, contact information
Process	Provide tools for adding, updating, or removing sponsor profiles. Manage and maintain contact information for sponsors.
Output	Well-organized and up-to-date sponsor profiles with accurate contact information.
Definition	facilitates the management of sponsor profiles and contact information. The system includes tools for adding, updating, or removing sponsor profiles and ensures the accurate maintenance of contact details.

F19	Sponsorship package management
Input	Details about different sponsorship package options, including benefits and pricing.
Process	Provide a platform to create and manage various sponsorship packages. Enable customization of benefits, pricing, and terms for each package.
Output	Diverse sponsorship packages with listed benefits
Definition	facilitates the management of sponsorship packages by providing a platform to create and customize various options.

F20	Track sponsor commitments and deliverables
Input	Sponsorship agreements, deliverable details
Process	Tracking system for commitments and deliverables
Output	Managed commitments and delivered sponsor benefits
Definition	Monitoring and managing sponsor obligations and outcomes

ii. Sponsors

F21	Sponsorship Package Information Access
Input	Package details, benefits information
Process	Provide a user-friendly interface for sponsors to access detailed information on available sponsorship packages and their associated benefits.
Output	Detailed information about available sponsorship packages
Definition	allows sponsors to securely access detailed information about available sponsorship packages and their associated benefits. Using a user-friendly interface, authenticated sponsors can review and explore the features, pricing, and advantages of each sponsorship package, ensuring informed decision-making.

F22	Materials and Logos Submission for Promotion
Input	Sponsorship materials, logo submissions
Process	Provide a user-friendly platform for users to submit required promotional materials and logos. Validate and process submitted materials according to specified formats
Output	Successfully submitted and validated promotional materials/logos for use in promotional activities

Definition	designed for the submission of promotional materials and logos. The output is a collection of successfully submitted and validated promotional materials/logos ready for use in promotional activities associated with the event.

3.4 Payment and Financial Management

i. Event Organizers

F23	Integrate secure payment gateways
Input	Payment gateway integration details
Process	Integration process with secure payment gateways
Output	Integrated and secure payment systems for ticket sales
Definition	Integration of secure payment channels for ticket sales

F24	Payment transaction processing
Input	Payment details, including user transactions and sales information.
Process	Real-time tracking system for financial transactions
Output	Real-time overview of financial transactions and revenue
Definition	manages transaction data and sales details by securely capturing and storing relevant information. The system offers tools for users to efficiently manage and analyze sales details, resulting in well-organized records and comprehensive insights into the sales performance.

ii. Financial Team

F25	Financial report generation
Input	Financial data, transaction records, and relevant parameters.
Process	Automatically generate comprehensive financial reports based on the provided financial data. Include options for customizing report parameters and formats
Output	Detailed and customizable financial reports.
Definition	automates the generation of comprehensive financial reports by utilizing provided financial data. Users can customize report parameters and formats as needed, resulting in detailed and customizable financial reports for effective analysis and decision-making.

F26	Invoice and payment management
Input	Invoice data, payment details
Process	Facilitate the creation, organization, and tracking of invoices. Provide tools for users to manage payments, including tracking and updating payment statuses
Output	Well-organized invoices and comprehensive records of payments.
Definition	streamlines the management of invoices and payments by facilitating the creation, organization, and tracking of invoices. Financial team can utilize tools to efficiently manage payments, including tracking and updating payment statuses, resulting in well-organized invoices and comprehensive records of payments.

F27	Ensure compliance with financial regulations
Input	Regulatory guidelines, compliance requirements
Process	Compliance checks, adherence mechanisms
Output	Compliance with financial regulations and guidelines
Definition	Ensuring adherence to financial regulations and standards

iii. Registration Team

F28	Payment status monitoring
Input	Registration and payment details of event attendees.
Process	Systematically track and monitor the payment status of registrations. Provide real-time updates on payment processing, pending payments, and successful transactions.
Output	Clear and up-to-date information on the payment status for each registration.
Definition	Tracking and updating the status of payments for registrations

F29	Payment issue resolution
Input	User-reported payment problems or discrepancies.
Process	Provide a dedicated support system to address and resolve payment-related issues, implement a streamlined process for investigating and resolving reported problems.
Output	Timely and effective resolution of payment-related issues.
Definition	ensures the timely resolution of payment-related issues by providing users with a dedicated support system. This feature involves a streamlined process for investigating and addressing reported problems, ensuring effective resolution and user satisfaction.

4. External Interface Requirements

4.1 User Interfaces

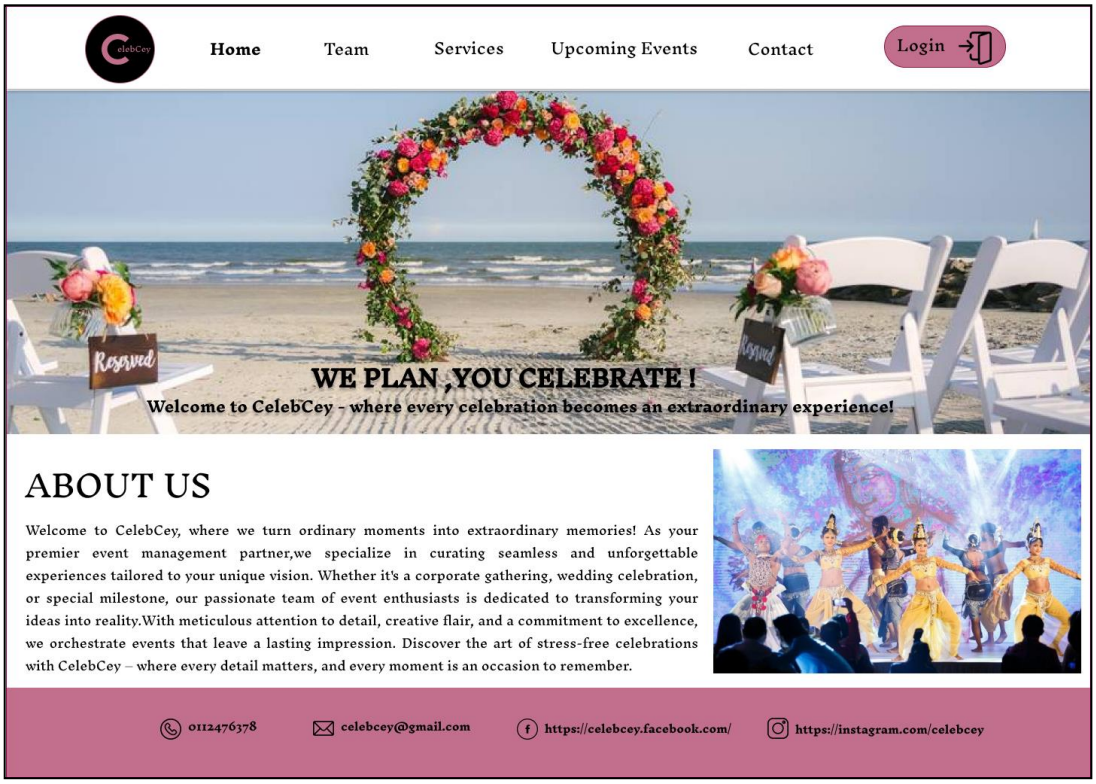

Figure 3: Home page

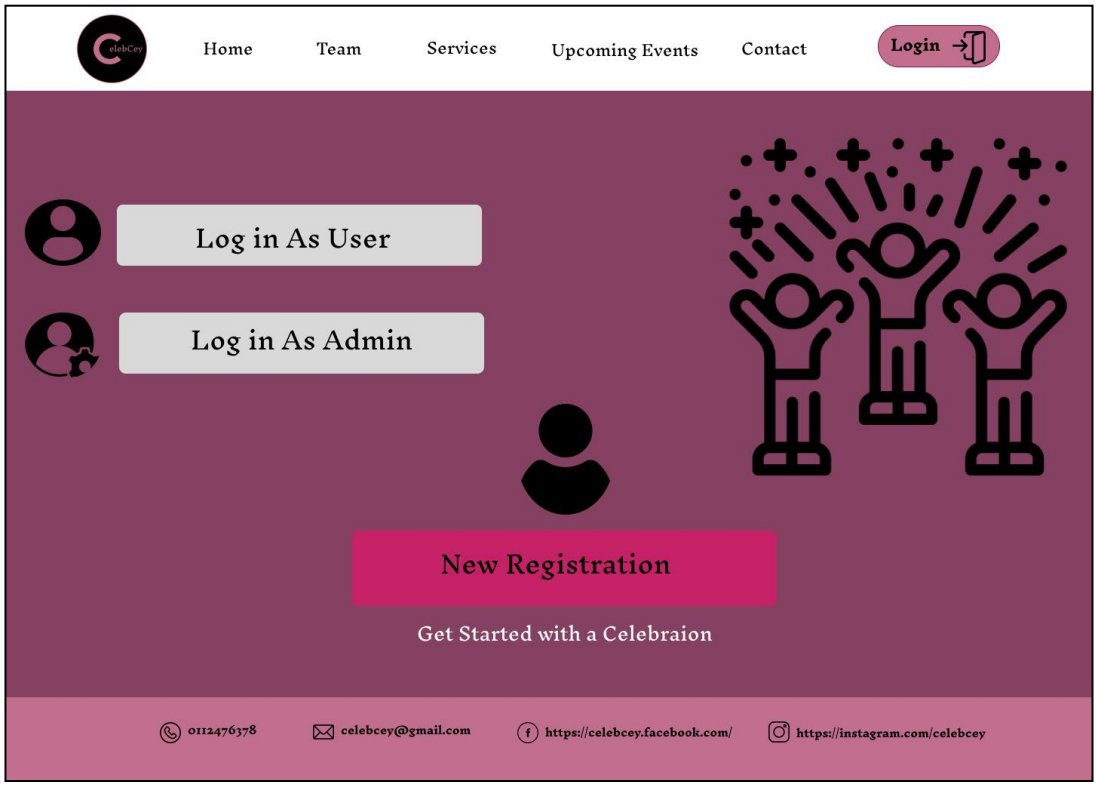

Figure 4: Login page

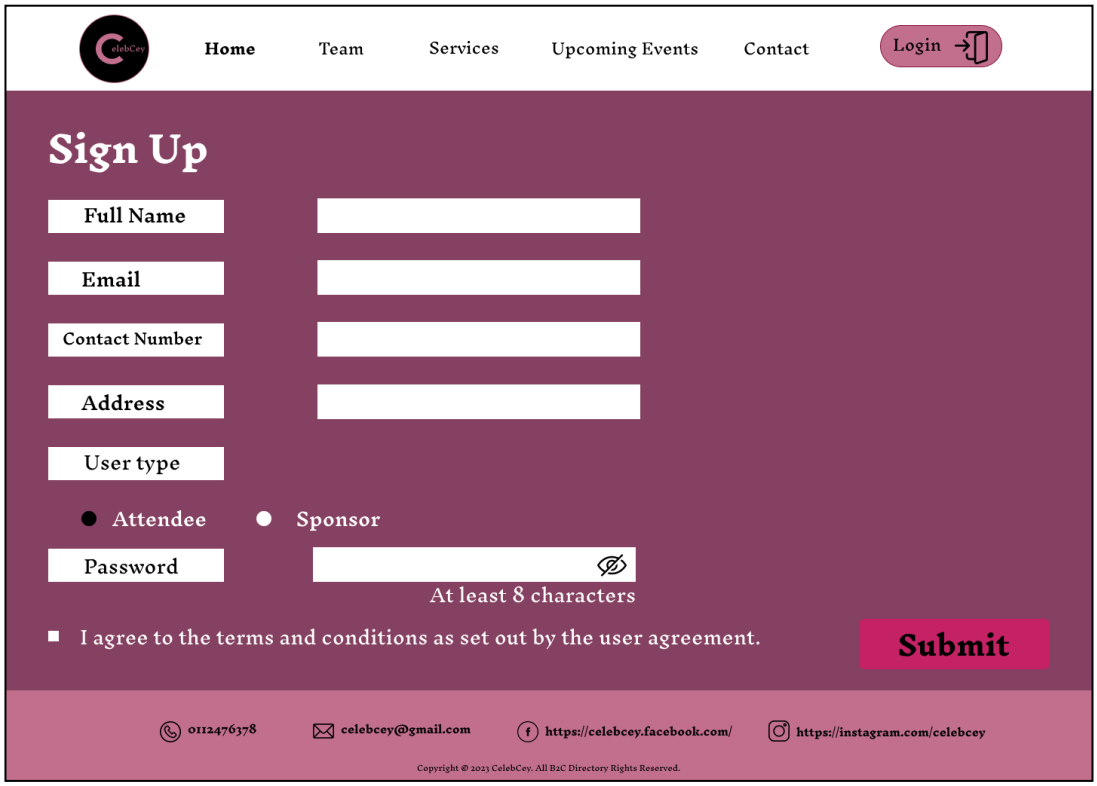

Figure 5: Registration page

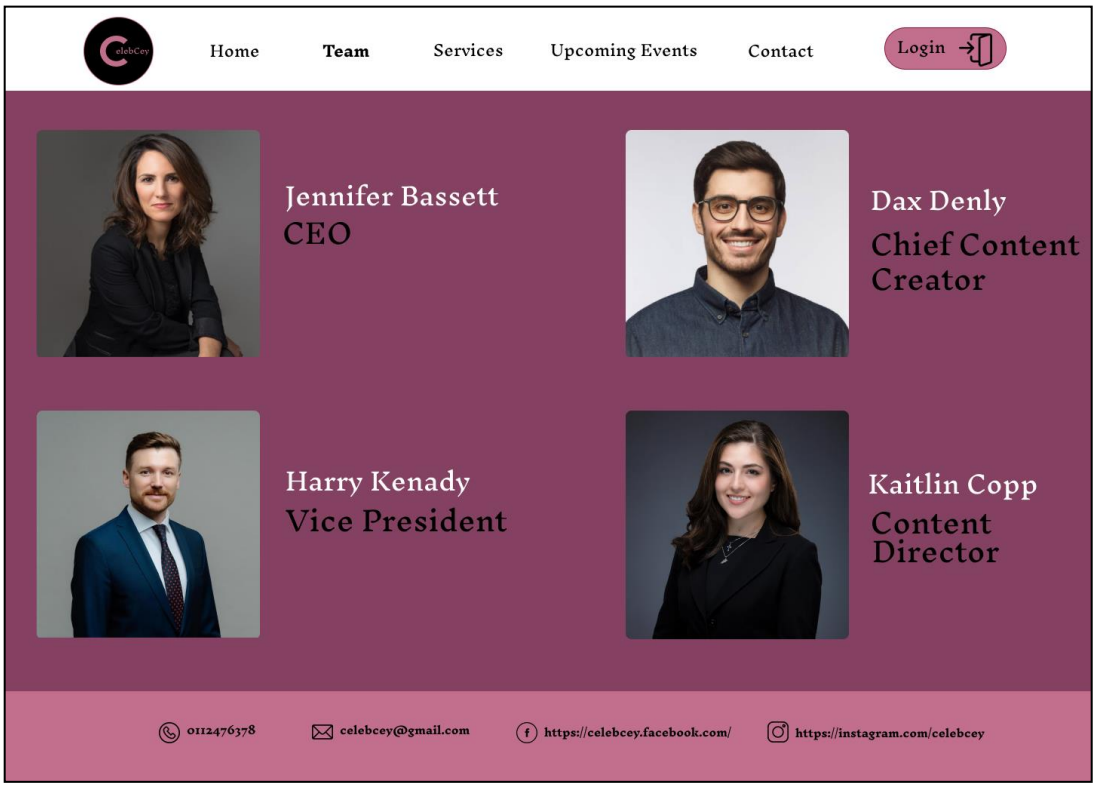

Figure 6: Team page

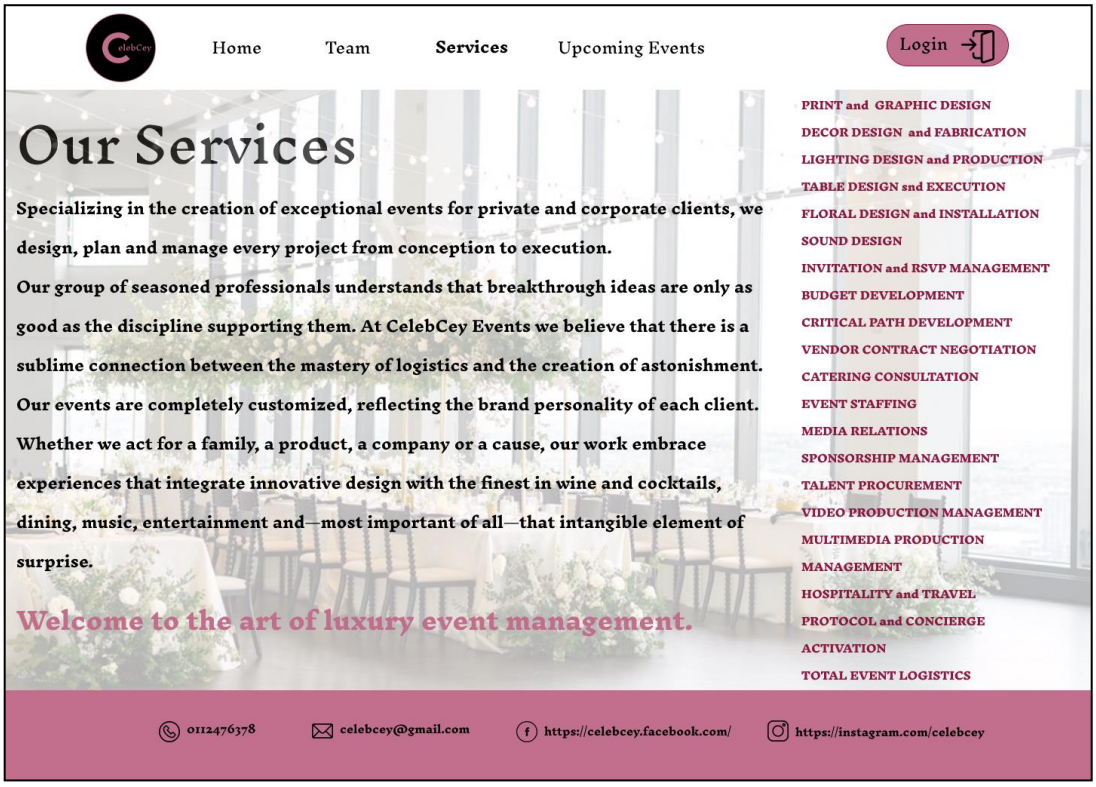

Figure 7: Services page

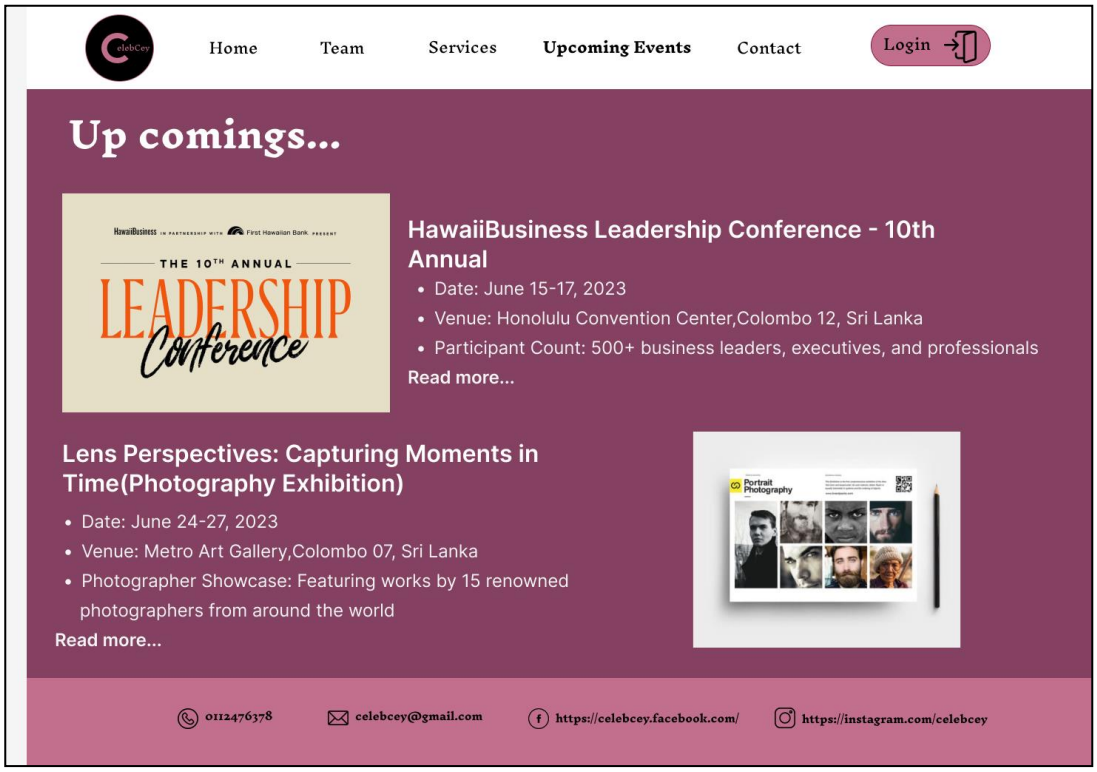

Figure 8: Upcoming event page

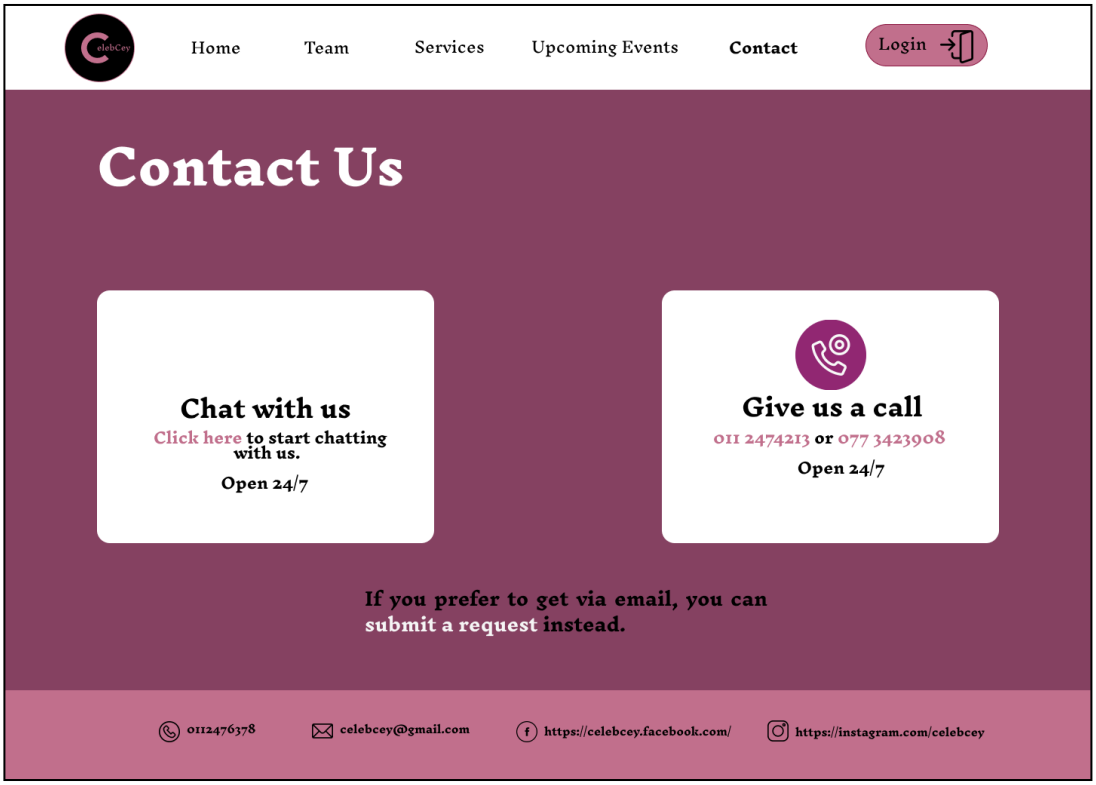

Figure 9: Contact Page

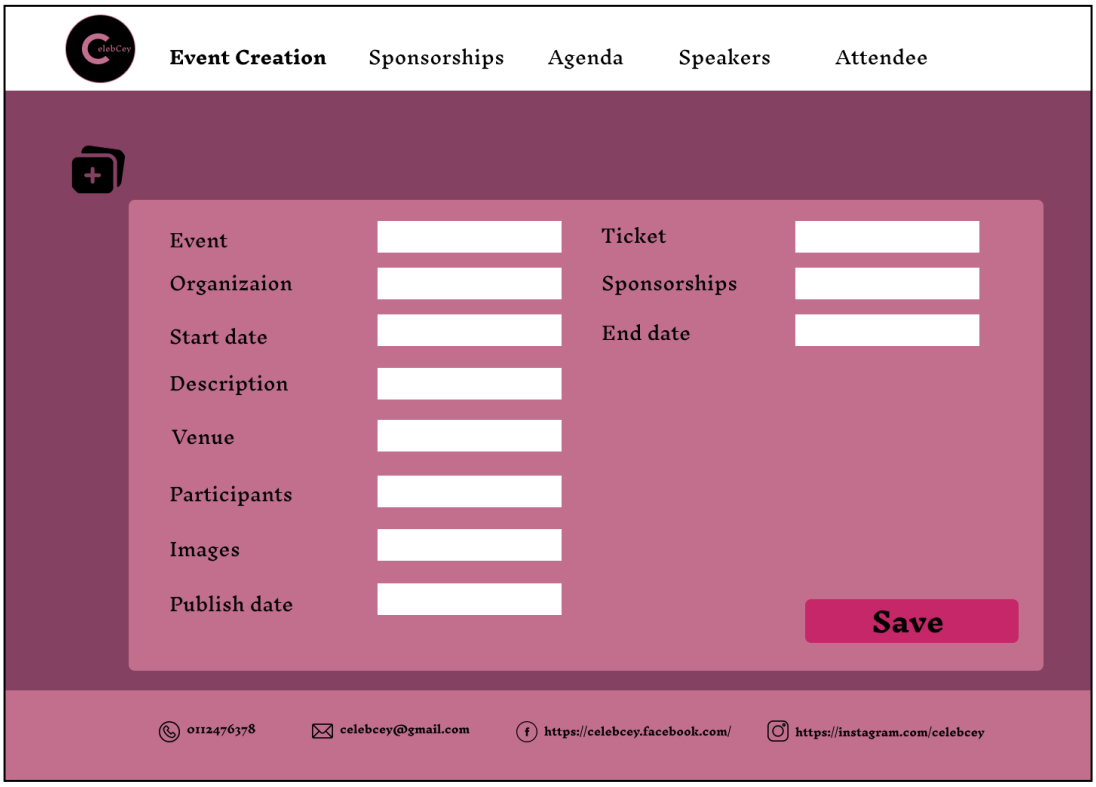

Figure 10: Interface for event organizers

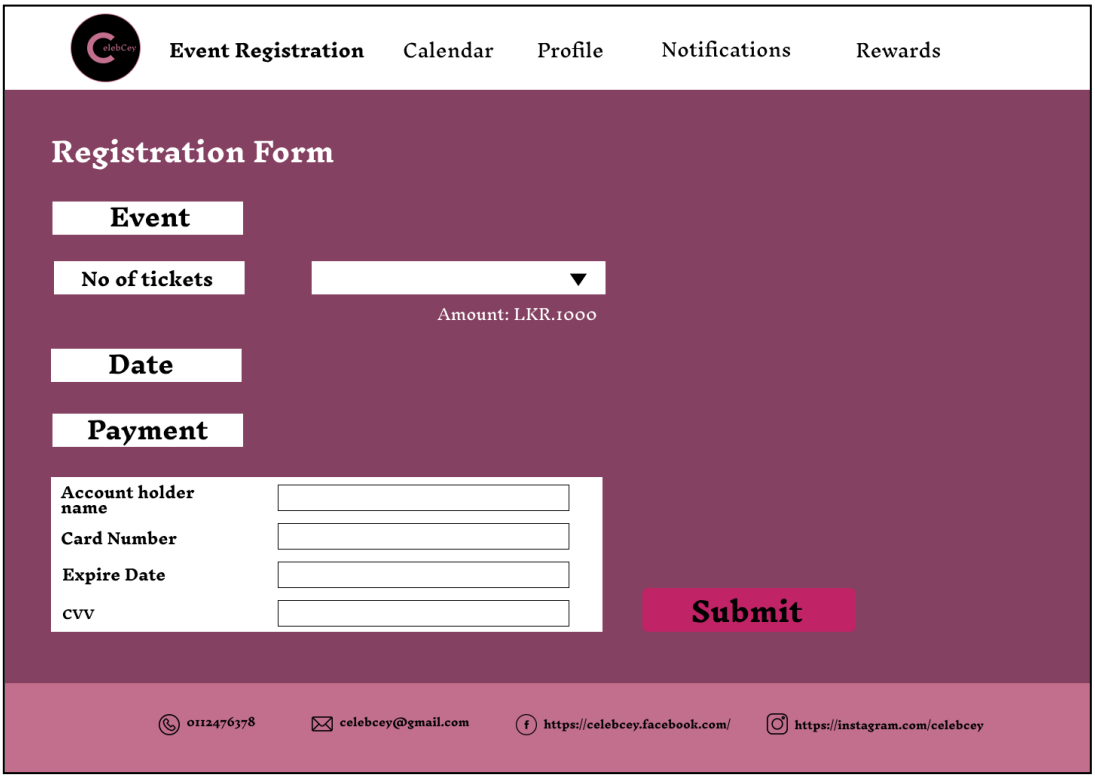

Figure 11: Interface for Attendees

Figure 12: Interface for sponsors

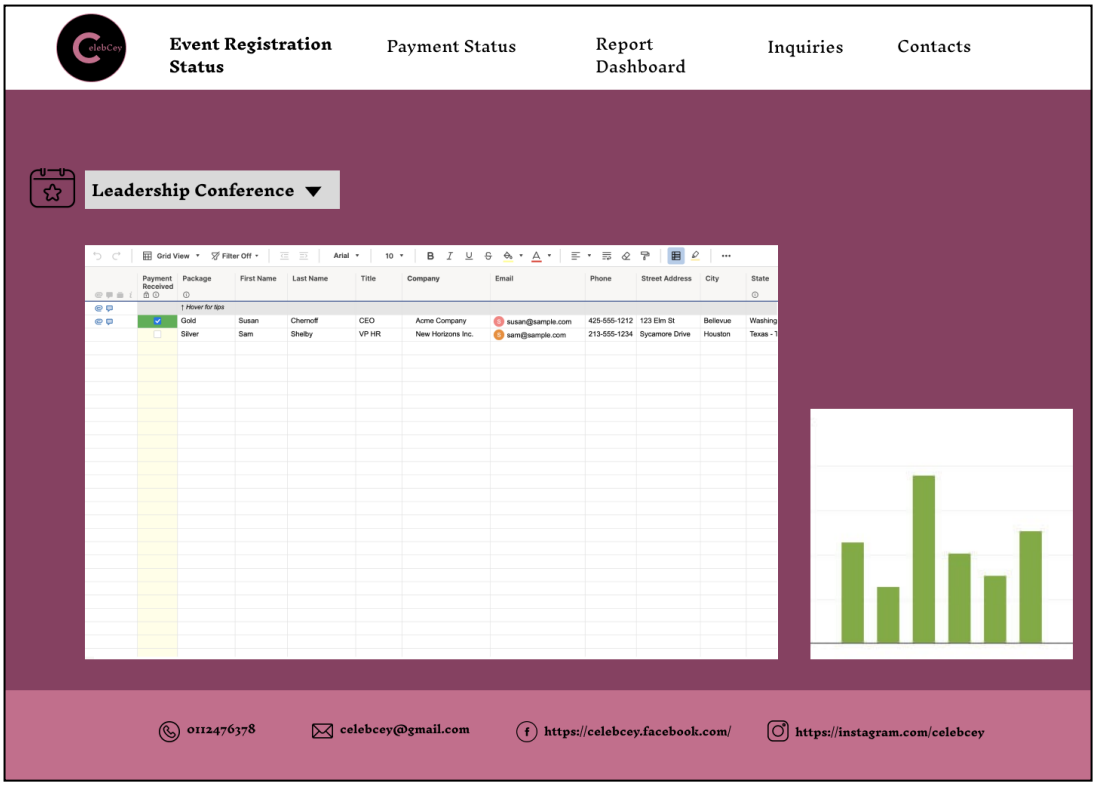

Figure 13: Interface for event registration team

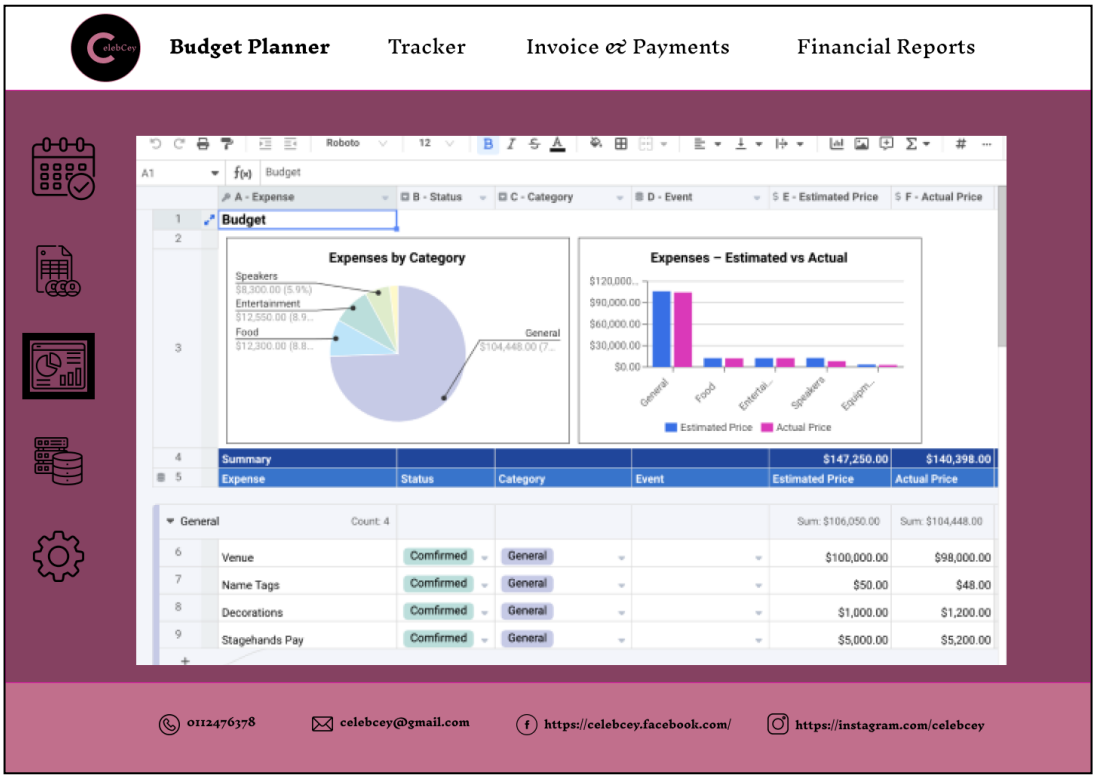

Figure 14: Interface for financial team

4.2 Hardware Interfaces

1. Supported devices.
    - Desktops/Laptops
        - ✓ Minimum RAM: 4GB.
        - ✓ Storage: 4 GB.
        - ✓ Processor: intel core/AMD or equivalent.
        - ✓ OS: Windows/MAC OS or equivalent.
    - Tablets/Smartphones
        - ✓ Minimum RAM: 4GB.
        - ✓ Storage: 4GB.
        - ✓ Processor: Equivalent android/ios processors.
2. Data and control interactions
    - Input devices
        - ✓ Standard keyboards, mics, and touchpad devices.
        - ✓ Touchscreen for tablets and smartphones.
    - Output devices
        - ✓ Monitor or screen.
        - ✓ Audio output support for system notifications
3. Communication protocols
    - Network model
        - ✓ Client-Server model.
        - ✓ Uses HTTP/HTTPS protocols for communication.

4.3 Software Interfaces

Interfaces	Connections	Data items/Messages	Purpose
1.Database Interface (e.g. MySQL version 4.4):	Interfaces with a MongoDB database to store and retrieve event-related data.	-Incoming: MongoDB queries for data retrieval, insertion, and modification. -Outgoing: Query results, transaction status updates.	To store and manage event details, user information, and transaction records.
2.Payment Gateway Interface (e.g. Stripe API version 3.0):	Integrates with the payment gateway for processing online payments.	-Incoming: Payment requests with transaction details. -Outgoing: Payment confirmation, transaction status.	To securely handle online transactions and provide real-time payment updates.
3.Authentication Service (e.g. OAuth 2.0):	User authentication and authorization.	-Incoming: User login requests with credentials. -Outgoing: Authentication tokens, user authorization	To ensure secure and authorized access to the EMS.

		status.	
4.Notification Service (e.g. Twilio API version 5.0):	Sending event-related notifications via SMS or email.	-Incoming: Notification requests with recipient details and content. -Outgoing: Delivery status, confirmation messages.	To inform users about event updates, ticket confirmations, and other relevant information.
5.Mapping Service (e.g. Google Maps API version 4.0):	Integrates with the Google Maps API for location-based services.	-Incoming: Location details, map rendering requests. -Outgoing: Map images, geolocation data.	To display event venues, generate maps, and provide location-based information.
6.Event Calendar Integration	Supports calendar format for event scheduling and synchronization.	-Incoming: Event details in calendar format. -Outgoing: Event updates, reminders.	To enable users to synchronize events with external calendar applications.
7.Ticketing System API (e.g. Ticketmaster API version 2.0):	Interfaces with the Ticketmaster API for additional ticketing services.	-Incoming: Ticket availability requests, pricing details. -Outgoing: Ticket purchase confirmations, inventory updates.	To provide users with additional ticketing options and integrate with external ticketing platforms.
8.Analytics and Reporting (e.g. Google Analytics):	for tracking user interactions and generating reports.	-Incoming: User activity data, page views. -Outgoing: Analytics reports, performance metrics.	To analyse user behaviour, track system performance.

4.4 Communications Interfaces

To enable efficient and secure connection between the system and external components, communication interfaces establish protocols, standards, security measures, and data formats. They also define the needs associated with specific functions required by the Event Management System.

Interfaces	Requirements
1.Web Browser Communication:	Users interact with the system through web browsers (Chrome, Firefox, Safari). Utilizes HTTP/HTTPS protocols for communication.
2. Email Communication:	Sends email notifications for event updates, ticket confirmations, and reminders. Uses SMTP (Simple Mail Transfer Protocol) for outgoing email communication.
3. Network Server Communication:	Communicates with the database server for data retrieval and storage. Uses TCP/IP for reliable data transmission.

	Server-to-server communication.
4. Payment Gateway Communication:	Interfaces with the payment gateway for secure transaction processing. Uses HTTPS to encrypt payment data during transmission.
5. Security and Encryption:	Implements TLS (Transport Layer Security) for secure communication across all interfaces. Ensures end-to-end encryption for sensitive data, such as user credentials and payment information.
6. Data Transfer Rates and Synchronization:	Defines acceptable data transfer rates to ensure responsive user interactions. Implements synchronization mechanisms for real-time updates, especially in the case of ticket availability and event changes.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

1. System Response Time:

   - ✓ Requirement: The system should respond to user interactions within 2 seconds under normal load conditions.
   - ✓ Description: Ensuring a prompt response time enhances user satisfaction and usability.

2. Concurrent User Handling:

   - ✓ Requirement: The system must support a minimum of 1500 concurrent users without significant degradation in performance.
   - ✓ Description: Supporting concurrent users is critical for scalability, especially during peak usage times.

3. Timing Relationships:

   - ✓ Requirement: For real-time features, such as live agenda updates, the system must maintain synchronization with a latency of less than 500 milliseconds.
   - ✓ Description: Real-time synchronization ensures that users receive up-to-date information and have a seamless experience.

4. Load Testing:

   - ✓ Requirement: Conduct load testing to simulate a usage scenario with 2000 concurrent users to ensure performance stability.
   - ✓ Description: Load testing helps identify potential bottlenecks and performance issues under high user loads.

5. Resource Utilization:

   - ✓ Requirement: The system should efficiently utilize system resources, with CPU utilization below 70% and memory usage below 80% during peak usage.
   - ✓ Description: Monitoring resource utilization helps prevent system overloads and ensures consistent performance.

6. Data Caching:

   - ✓ Requirement: Implement data caching mechanisms to minimize redundant database queries and improve response times.
   - ✓ Description: Caching frequently accessed data helps optimize system performance.

7. Error Handling Time:

   - ✓ Requirement: The system should display error messages within 5 seconds of encountering an issue, providing clear and actionable information.
   - ✓ Description: Swift error handling contributes to a positive user experience and facilitates issue resolution.

8. Scalability:

   - ✓ Requirement: The system architecture should be designed to scale horizontally to accommodate a potential 50% increase in user base within the next year.
   - ✓ Description: Scalability ensures the system can handle growth without compromising performance.

9. Continuous Monitoring:

   - ✓ Requirement: Implement continuous monitoring tools to track system performance metrics and identify potential issues proactively.
   - ✓ Description: Ongoing monitoring helps maintain optimal performance and allows for quick identification and resolution of any issues.

5.2 Safety Requirements

1. Data Security:

   - ✓ Requirement: Implement robust data security measures to protect user information, including personal details and financial transactions.
   - ✓ Safeguards: Encryption of sensitive data during transmission and storage, regular security audits, and compliance with data protection regulations. Use Transport Layer Security (TLS) for encrypting data in transit and implement strong encryption algorithms (e.g., AES) for storing sensitive user information in the database.
   - ✓ Description: Ensures the confidentiality and integrity of user data, preventing unauthorized access or data breaches.

2. Payment Security:

    - ✓ Requirement: Adhere to Payment Card Industry Data Security Standard (PCI DSS) for processing and storing payment information securely.
    - ✓ Safeguards: Encryption of payment data, secure payment gateway integration, and regular PCI compliance assessments. Utilize a PCI-compliant payment gateway (e.g., Stripe) with end-to-end encryption to secure payment transactions and comply with industry standards.
    - ✓ Description: Mitigates the risk of financial loss and protects users' sensitive payment information.

3. User Authentication:

    - ✓ Requirement: Implement a secure user authentication mechanism to prevent unauthorized access to user accounts.
    - ✓ Safeguards: Use of strong password policies, multi-factor authentication, and account lockout mechanisms. Enforce password complexity requirements, implement two-factor authentication using SMS or authentication apps, and lock user accounts after a specified number of failed login attempts.
    - ✓ Description: Safeguards against unauthorized access, protecting user accounts and ensuring data privacy.

4. Error Handling:

    - ✓ Requirement: Provide clear and user-friendly error messages to guide users in the event of system errors or issues.
    - ✓ Safeguards: Avoid displaying sensitive information in error messages, log error details securely, and prompt users to contact support for assistance. Display a generic error message to users while logging detailed error information in a secure log file accessible only to authorized personnel for debugging purposes.
    - ✓ Description: Prevents confusion, enhances user understanding, and helps users take appropriate actions in response to errors.

5. Backup and Recovery:

    - ✓ Requirement: Implement regular backup procedures to ensure data integrity and facilitate system recovery in the event of data loss or system failures.
    - ✓ Safeguards: Automated backup schedules, secure storage of backup data, and periodic recovery testing. Schedule daily automated backups of the database, store backups in a secure offsite location, and regularly perform recovery drills to validate the effectiveness of the backup and recovery process.
    - ✓ Description: Mitigates the impact of data loss, ensuring the availability and reliability of the system.

6. Compliance with Accessibility Standards:

    - ✓ Requirement: Design the system to comply with accessibility standards (e.g., WCAG) to ensure usability for users with disabilities.
    - ✓ Safeguards: Use of accessible design principles, testing with assistive technologies, and adherence to accessibility guidelines. Provide alternative text for images, ensure keyboard navigation, and use semantic HTML to enhance accessibility for users with visual or motor impairments.

- ✓ Description: Ensures inclusivity and prevents harm by providing equal access to all users.

7. Emergency Notifications:

- ✓ Requirement: Implement emergency notification features to communicate critical information to users in case of unexpected events or changes to events.
- ✓ Safeguards: Clearly communicate emergency information, ensure timely notifications, and provide instructions for users to take appropriate actions. Send push notifications to users' devices in case of event cancellations, changes in schedule, or emergency situations, accompanied by clear instructions on what actions users should take.
- ✓ Description: Enhances user safety by providing timely and accurate information during emergencies.

5.3 Security Requirements

1. Data Privacy:

- ✓ Requirement: Protect user privacy by adhering to data protection laws and regulations (e.g., GDPR). Obtain explicit user consent before collecting and processing personal information. Allow users to review and update their privacy preferences, including opting out of certain data processing activities.
- ✓ Description: Implement features such as data anonymization, obtain user consent for data processing, and provide transparent privacy policies.

2. Security Training:

- ✓ Requirement: Provide security awareness training for personnel involved in the development and operation of the system. Conduct regular security awareness training for employees, covering topics such as recognizing phishing attempts, practicing secure password management, and reporting security incidents promptly.
- ✓ Description: Train employees on security best practices, social engineering awareness, and the importance of safeguarding sensitive information.

3. Security Certifications:

- ✓ Requirement: Obtain and maintain relevant security certifications (e.g., ISO 27001, SOC 2) to demonstrate the system's commitment to security. Work towards achieving ISO 27001 certification, demonstrating the implementation of a comprehensive information security management system. Regularly undergo audits to maintain certification status.
- ✓ Description: Work towards achieving and renewing industry-recognized security certifications, providing assurance to users and stakeholders.

4. Password Policies:

- ✓ Requirement: Enforce strong password policies to enhance the security of user accounts. Enforce a password policy requiring a minimum of 10

characters, including at least one uppercase letter, one lowercase letter, one digit, and one special character. Prompt users to update their passwords every 90 days to enhance security.
- ✓ Description: Specify minimum password length, complexity requirements, and periodic password changes to prevent unauthorized access.

5.4 Software Quality Attributes

1. Usability:
   - ✓ Requirement: Achieve a System Usability Scale (SUS) score of at least 75.
   - ✓ Description: Conduct usability testing to ensure that users find the system easy to use and navigate. Use SUS scores as a quantitative measure of usability.

2. Reliability:
   - ✓ Requirement: Maintain a system uptime of at least 99.5%.
   - ✓ Description: Implement robust error handling, conduct stress testing, and use automated monitoring tools to ensure high system reliability and availability.

3. Maintainability:
   - ✓ Requirement: Achieve a code maintainability index score of 80 or higher.
   - ✓ Description: Follow coding standards, utilize version control, and document code comprehensively to facilitate ease of maintenance and future enhancements.

4. Portability:
   - ✓ Requirement: Ensure the system runs seamlessly on major web browsers (Chrome, Firefox, Safari) and operating systems (Windows, macOS, Linux).
   - ✓ Details: Utilize web standards and responsive design principles to enhance cross-browser and cross-platform compatibility.

5. Performance Efficiency:
   - ✓ Requirement: Load web pages within 3 seconds under standard network conditions.
   - ✓ Description: Optimize code, utilize content delivery networks (CDNs), and implement caching mechanisms to enhance overall system performance.

6. Testability:
   - ✓ Requirement: Achieve a test coverage rate of 80% or higher.
   - ✓ Details: Develop comprehensive test cases, utilize unit testing and integration testing, and automate testing processes to ensure thorough test coverage.

7. Adaptability:
   - ✓ Requirement: Support regular updates and feature additions without disrupting existing user experiences.
   - ✓ Description: Implement a modular architecture, use feature flags for controlled releases, and conduct A/B testing for new features.

8. Flexibility:
   - ✓ Requirement: Allow users to customize event notification preferences and dashboard layouts.

- ✓ Description: Implement user-configurable settings, providing flexibility in tailoring the system to individual preferences.

9. Availability:
   - ✓ Requirement: Maintain 24/7 availability for critical system functionalities, such as event registration and ticketing.
   - ✓ Description: Implement redundancy, utilize cloud hosting with high availability, and conduct regular maintenance during non-peak hours.

10. Robustness:
    - ✓ Requirement: Handle a concurrent user load of 2000 users without critical system failures.
    - ✓ Description: Conduct load testing, implement graceful error recovery mechanisms, and ensure system components are resilient under heavy usage.

6. Other Requirements

6.1 Accessibility Requirements

The system shall adhere to accessibility standards (e.g., WCAG) to ensure that individuals with disabilities can use the software effectively. User interfaces shall be designed to accommodate various assistive technologies.

7. Appendix A: Analysis Models

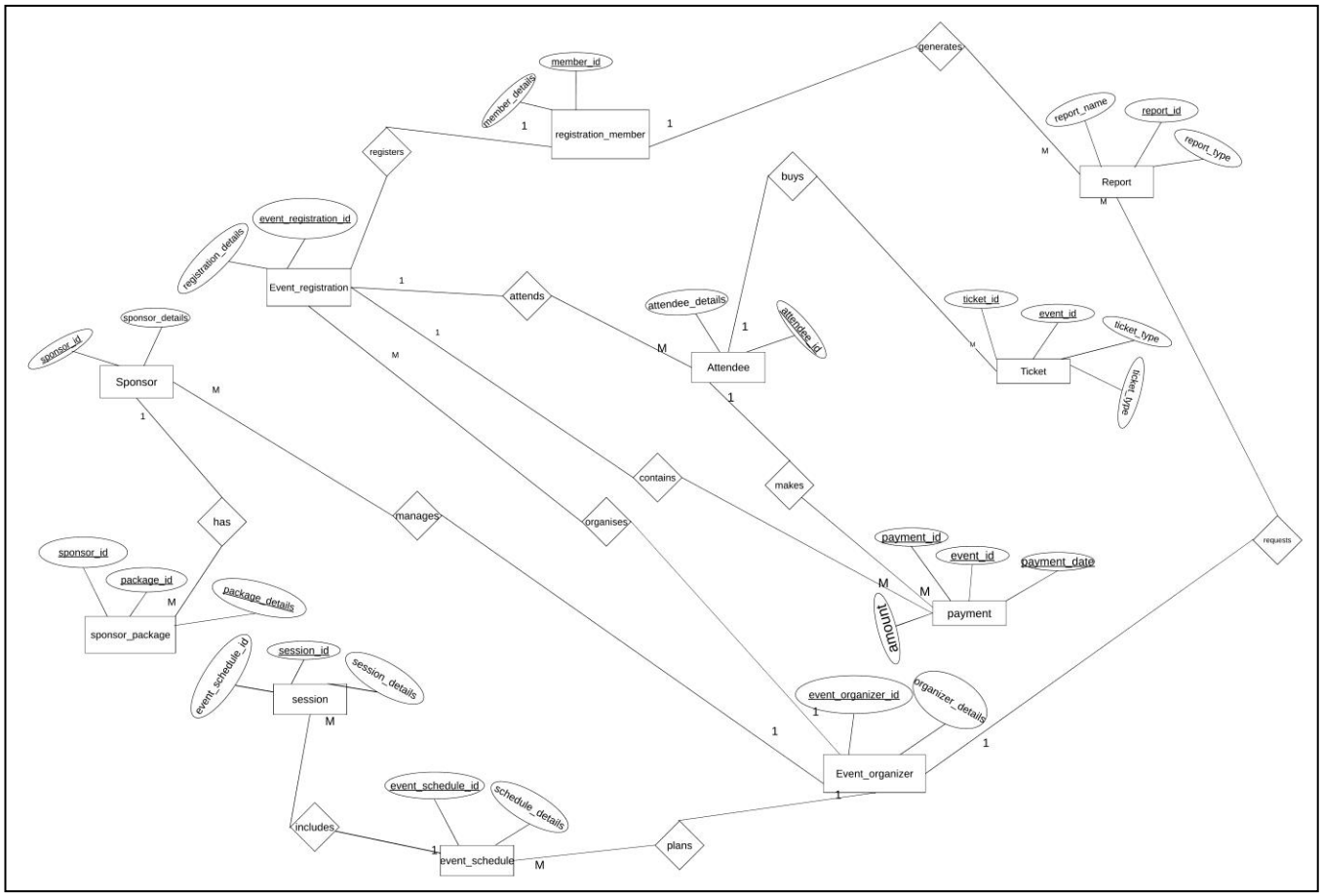

Figure 15: Entity relational diagram