Thapar Event Management

UCS503 Software Engineering Project Report

Submitted by:

(102203775) Radhika

(102203794) Manvi

(102203968) Khushi

BE Third Year, COE

Computer Science and Engineering Department

TIET, Patiala December 2024

TABLE OF CONTENTS

`

S. No.	Topic	Page No.
1	Software Bid form	3
2	Project Description	6
3	Feasibility Report	7
4	Gantt Chart	10
5	Software Requirement Specification (SRS)	11
6	Use-Case Diagram	18
7	DFD –Level0	19
8	DFD –Level1	20
9	Activity Diagram	21
10	Sequence Diagram	22

Software Bid/ Project Teams

UCS 503- Software Engineering Lab

Group : 3CO32 Dated: 22-08-2024

Team Name: Error 404
Team ID (will be assigned by Instructor):

Please enter the names of your Preferred Team Members. :

· You are required to form **a three to four person** teams

· Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used	Signature
Radhika Rani	102203775	Thapar Satellite Project, Sudoku Practice Website	HTML, CSS , JS, C++, Python	
Manvi Bansal	102203794	Login Portal, Budget Calculator	HTML, CSS, JS	
Khushi Bansal	102203968	To-Do List, Voice Assistant, Toxic Comment Detection	Django, Python, ML , DL	
Vaibhav	102203779	Online Food Delivery Management System	MySQL, HTML, CSS, JS	

Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. FrontEnd : HTML, CSS, JS(React / Angular)
2. BackEnd: Django/ Node.js , Express.js
3. MongoDB
4. Coding Languages known: C++, Python

Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference: *[Write at-least one paragraph for each choice (motivation, reason for choice, feasibility analysis, etc.)]*

	Project Name	Unique Selling Point
First Choice	Thapar Event Management Website	Say goodbye to long queues and missed opportunities. Our website offers seamless online ticket booking and instant event access, ensuring that every student enjoys a hassle-free and unforgettable college journey. **Features:** 1. Instant ticket booking in a few clicks.

		2. Fast event entry with digital tickets and QR codes. 3. Stay updated with reminders and alerts. 4. Receive personalized recommendations.
Second Choice	Project Collaboration Website for Professors and Students	Connecting Professors and Students Across Universities for Targeted Research Collaboration, Without the Need for Physical Visits or Endless Emails. **Features:** 1. Filtered Opportunities 2. Cross-Institutional Collaboration 3. Transparent Project Listing 4. Streamlined communication 5. Resource Hub 6. Community Building
Third Choice	Thapar Map Website	Navigate Thapar University with Precision: Your Personalized Guide to Every Corner. Discover seamless navigation from your current location to any destination on campus, with detailed insights on shops, libraries, cafeterias, and more. Tailored for ease, accuracy, and real-time updates, making every visit effortless and efficient. **Features:** 1. Precision Navigation 2. Comprehensive Insights 3. Seamless User Experience 4. Real-Time Updates
Fourth Choice	Blood Bank Management	Revolutionizing Blood Management with Real-Time Accuracy and Seamless Connectivity: A Comprehensive Platform for Donors, Blood Banks, and Hospitals. **Features:** 1. Real-Time Inventory Management 2. A unified platform that eliminates data fragmentation, ensuring accuracy and accessibility of information. 3. Simplified donor management with personalized reminders, health tracking, and easy scheduling. 4. Built-in compliance checks and digital documentation to meet regulatory requirements effortlessly.

Additional Remarks/ Inputs

Please tell us about any other factors that we should take into consideration (e.g., if you really would like to work on a project for some particularly convincing reason).

The idea for the Thapar Event Management Website was inspired by our experience with long queues for event passes, particularly at the Thapar Food Festival and Frosh events. I want to eliminate these hassles by providing instant online ticket booking, digital tickets with QR codes for fast entry, timely reminders, and personalized event recommendations. This project aims to enhance the overall student experience by making event participation seamless and enjoyable.

Project Description

The **Thapar Event Management Website** is a user-friendly platform developed to streamline event management at Thapar University. The website allows students, staff, and event organizers to manage and participate in events with ease. Key features include instant online ticket booking, digital tickets for fast event entry, and personalized event recommendations based on user preferences. The platform also sends reminders and alerts to ensure users stay informed about upcoming events.

This project was motivated by the challenges faced during events like the Thapar Food Festival, where long queues and manual processes caused significant delays and inconvenience. By digitizing the entire ticketing and event management process, the platform enhances efficiency, making event participation more accessible and enjoyable for all users.

Technically, the platform will be built using modern web development technologies such as HTML, CSS, and JavaScript (React) for the frontend, while the backend will be powered by Node.js with Express.js. Data will be securely stored and managed using databases like MongoDB. The website will also integrate secure payment gateways for ticket purchases and use e-ticket for instant, secure event access.

Targeting the entire university community, the platform aims to be scalable, supporting thousands of users during peak event times. It will also be optimized for secure and reliable operations, ensuring smooth functionality during high traffic periods. The Thapar Event Management Website is designed to simplify the event experience, making it a more seamless and efficient process for everyone involved.

Project Feasibility Analysis

Motivation: The project aims to address the inconvenience of long queues and missed opportunities at university events. By digitizing the ticketing process, the project seeks to create a more efficient and enjoyable event experience.

1. Technical Feasibility

1.1. Software Requirements

Hardware

- **Servers**: Servers are needed to host the website, with enough power to handle busy times during major events.
- **Storage Systems**: Storage is required to save user data, event info, and digital tickets.

Software Stack

- **Frontend**: The user interface will be built with HTML, CSS, and JavaScript frameworks like React.js or Angular.
- **Backend**: The server-side code will use Node.js with Express, Django, or Ruby on Rails.
- **Database**: Data will be stored in databases like MySQL, MongoDB, or PostgreSQL.
- **APIs**: Integration of payment systems and tools for generating QR codes.

1.2. **Other Technical Feasibility**

- Access needed for the student database for personalizing event recommendations.
- Ensuring the event information is accurate and up-to-date.
- The platform will handle increased event traffic with load balancing to optimize performance and ensure smooth operation.
- Payments will be processed securely through compliant gateways, with secure authentication measures in place for users.
- Quality Assurance will test for functionality, performance, and security, while User Acceptance Testing will ensure the system meets users' needs.

2. Operational Feasibility

- **User Acceptance :**Provide clear tutorials and ongoing support while collecting and acting on user feedback during testing.
- **Workflow Compatibility**: Align the platform with existing university workflows to enable easy event management and smooth data exchange between users and the platform.
- **Accessibility and Availability***:*Ensure the website is supported on various devices and browsers, and maintain 24/7 availability, particularly during peak times.
- **Security**: Implement secure data storage and backup methods, and establish clear data management and protection policies.
- **Performance Optimization**: Plan for future growth in users and events, while regularly optimizing the platform's performance.

- **Maintenance**: Schedule regular software updates for new features and security enhancements, and provide help desk and troubleshooting support.

- **Cost-management**: Plan for development, hosting, and maintenance costs.

3. Economic Feasibility

- Estimate expenses for hiring developers, software tools, and server hosting.
- Account for ongoing costs like server maintenance, data storage, and user support.
- Include expenses for maintenance, updates, and staff salaries.
- Identify ways to earn revenue through partnerships with sponsors or vendors.
- Analyze potential revenue and estimate when the platform will become profitable.

4. Market Feasibility

- **User Base**: The platform targets the entire university community and event managers across all societies.
- **Demand**: There's strong demand for a centralized platform that consolidates events from all societies, providing a dedicated site for event managers to organize events.
- **Competition**: Limited competition within the university, as existing solutions are usually specific to individual societies.
- **User Engagement**: High engagement potential due to the platform's coverage of all societies' events, user-friendly interface, and tools for event managers.
- **Market Trends**: Growing preference for integrated digital solutions that simplify event management and address issues like overload, especially within educational institutions.

5. Scheduling Feasibility

- **Project Phases**: Set clear goals and deadlines for development, testing, and launch.

- **Essential Tasks:** Identify key tasks to keep the project on schedule.

- **Team Availability:** Ensure you have skilled developers, designers, and project managers.

- **Tools and Equipment:** Confirm access to necessary software and hardware.

- **Risk Planning**: Identify potential issues and prepare contingency plans.

- **Project Scope:** Define what the project will include and its features.

- **Change Management:** Establish a process for handling changes to the project.

- **Development Approach:** Use Scrum or Kanban for managing progress.

- **Incremental Progress:** Plan short, iterative sprints for continuous improvement.

6. Legal Feasibility

- **Intellectual Property:** Use content and software that are either owned, licensed, or legally authorized, avoiding copyright or trademark violations.

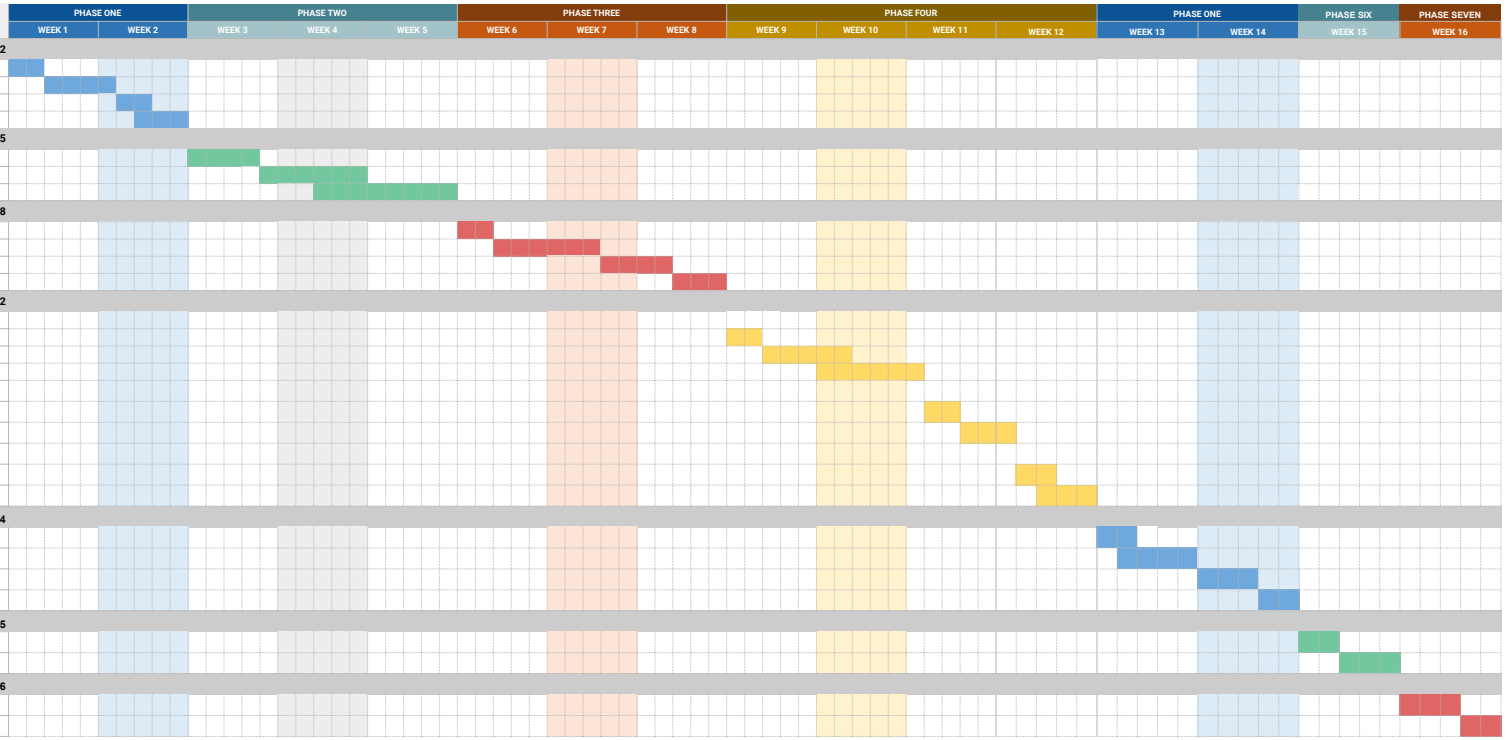
- **Accessibility:** Ensure accessibility and usability for individuals with disabilities.
- **University and Society Compliance:** All features and operations of the platform will comply with the legal and policy requirements of Thapar University and its affiliated societies.

Project Title: Event Management Web-App for Thapar University Date: 29/08/24

Team: Error404

Radhika Rani	102203775
Manvi Bansal	102203794
Khushi Bansal	102203968

WBS NUMBER	TASK TITLE	START DATE	DUE DATE	DURATION (in days)
1	**Project Conception**			**Week 1-2**
1.1	Define Project Scope	01/08/24	03/08/24	2
1.2	Assess Feasibility Factors	04/08/24	08/08/24	4
1.3	Project Planning	09/08/24	11/08/24	2
1.4	Project Initiation	12/08/24	15/08/24	3
2	**Analysis Phase**			**Week 3-5**
2.1	Use-Case Scenario of the project	16/08/24	19/08/24	4
2.2	Requirements and System Design	20/08/24	25/08/24	6
2.3	SRS Preparation	26/08/24	05/09/24	11
3	**Design Phase**			**Week 6-8**
3.1	Choose Tech-Stack	06/09/24	07/09/24	2
3.2	Develop UI/UX	08/09/24	13/09/24	6
3.3	System Design Architecture	14/09/24	16/09/24	3
3.4	ER-Diagram and Database design	17/09/24	19/09/24	2
4	**Coding Phase**			**Week 9-12**
4.1	**Frontend Development**			
4.1.1	Setup Project Structure	20/09/24	23/09/24	4
4.1.2	Develop Event Listing Page	24/09/24	30/09/24	7
4.1.3	Implement Ticket Booking Interface	01/10/24	08/10/24	10
4.2	**Backend Development**			
4.2.1	Set up Server and Database	09/10/24	13/10/24	5
4.2.2	Implement Ticket Booking logic	14/10/24	21/10/24	9
4.3	**Integration**			
4.3.1	Integrate Frontend and Backend	22/10/24	25/10/24	4
4.3.2	Test End-to-End Flow	26/10/24	30/10/24	5
5	**Testing Phase**			**Week 13-14**
5.1	Unit Testing	31/10/24	02/11/24	3
5.2	Integration Testing	03/11/24	06/11/24	4
5.3	User Acceptance Testing	07/11/24	10/11/24	4
5.4	Test Report Generation	11/11/24	14/11/24	3
6	**Deployment Phase**			**Week 15**
6.1	Set up Hosting Environment	15/11/24	17/11/24	3
6.2	Deployment and Documentation	18/11/24	21/11/24	4
7	**Review and Presentation**			**Week 16**
7.1	Monitor for Issues and Bugs	22/11/24	26/11/24	6
7.2	Project Submission and Presentation	27/11/24	28/11/24	1



A CASE STUDY (IEEE Format)

Software Requirements Specification Document

Thapar event management website

`

1. Introduction

1.1 Purpose of this Document

This document clearly describes the requirements for the Event Management Website, an online platform designed to help users plan and manage events. The website will allow users to create events, share details, and manage registrations. It explains the key features, user interface, and technical needs, ensuring that the client, development team, and users have a shared understanding of the platform's purpose and functions.

1.2 Scope of the Development Project:

The goal is to design a user-friendly Event Management Website for Thapar University. The website will enable students to easily book tickets for events, gain instant access using digital tickets and QR codes, receive event reminders and alerts, and get personalized event recommendations.

1. **Instant Ticket Booking**: Rapid ticket reservation through a user-friendly interface.
2. **Digital Ticket and QR Code Access**: Provide users with digital tickets and QR codes for fast entry to events.
3. **Reminders and Alerts**: Notify users about upcoming events and important updates.
4. **Personalized Recommendations**: Suggest events to users based on their interests and past activities.

1.3 Definitions, Abbreviations, and Acronyms

- **Event :-** A planned occasion or gathering where people come together for a specific purpose.
- **User:-**: An individual who interacts with the platform.
- **Admin:-** A person with authority who oversees the entire system.
- **Participant:-**A person who attends or registers for an event listed on the platform.

1.4 References

- IEEE SRS Case Study Template

1.5 Overview

The following sections provide a general description of the project, including system's architecture, user requirements, and technical specifications. Section 2 system's components, user interactions, and constraints, while Section 3 focuses on specific requirements, including external interfaces and detailed functional requirements.

`

2. Overall Description

2.1 Product Perspective

The Thapar Event Management Website will streamline event organization with online ticket booking, QR code access, and personalized recommendations. Accessible via basic devices with the internet, it allows users to log in with their roll number or admission number (for freshers) The platform provides real-time alerts and event browsing features.

2.2 Product Functions

1. **User Authentication**: Users will log in using their university-provided credentials. The system will authenticate the user and grant access to the appropriate features based on their role (student, staff, or event organizer).
2. **Event Browsing and Booking**: Users can browse upcoming events and book tickets online. The system will display event details, availability, and pricing, allowing users to make informed decisions.
3. **E-ticket for Event Access**: After booking, users will receive a digital ticket with a QR code that they can use for instant event access. The system will verify the QR code at the event entrance, ensuring secure and efficient entry.
4. **Personalized Event Recommendations**: The system will analyze users' past event participation and preferences to offer personalized recommendations for future events.
5. **Real-time Alerts and Reminders**: Users will receive reminders and alerts for upcoming events, ticket bookings, and other relevant notifications. The system will ensure users stay informed and prepared for their chosen events.
6. **Event Management for Organizers**: Organizers can create, edit, and manage events through the platform. They will have access to tools for setting ticket prices, managing attendee lists, and monitoring event status in real-time.
7. **Dashboard for Admins**: Admins will have access to a comprehensive dashboard that allows them to manage user profiles, monitor event participation, and oversee the overall functioning of the platform.

2.3 User Characteristics

1. **Students**: They will be the primary users, booking tickets and accessing events.
2. **Staff**: They will use the platform to organize events and manage bookings.
3. **Event Organizers**: They will create and manage events, track attendance, and engage with participants.
4. **Admins**: They will oversee the system, manage user roles, and ensure smooth operation.

`

2.4 General Constraints, Assumptions, and Dependencies

- **Responsive Design**: The website must be accessible across various devices (desktop, laptop, tablet, and mobile) with consistent performance and usability.
- **User Load**: The system must handle high volumes of users, particularly during peak event booking periods, with minimal downtime.
- **Security**: The platform must ensure secure user authentication, data privacy, and secure QR code generation to prevent unauthorized access.
- **Integration with University Systems**: The website should seamlessly integrate with existing university systems, such as the student information system for authentication and event notifications.
- **Data Backup and Recovery**: Regular backups of event data and user information must be maintained to ensure data integrity and availability in case of system failures.

3. Specific Requirements

3.1 External Interface Requirements

- **Home Page**: Shows featured events, a search bar, and navigation options.
- **Registration/Login Page**: Allows account creation or login with roll number or admission number.
- **Event Listing Page**: Enables organizers to add events with descriptions and ticket info.
- **User Profile Page**: This lets users manage accounts, view bookings, and track participation.
- **Screen Resolution**: Supports devices with resolutions from 320 x 240 pixel.

3.2 Detailed Description of Functional Requirements

3.2.1 Functional Requirements for Student Dashboard

Purpose	**The student dashboard provides personalized event recommendations, ticket booking options, and event reminders.**
Inputs	Students will input their login credentials to access personalized event information. They can browse events, book tickets, and view reminders using a simple, intuitive interface.
Processing	The platform will authenticate the student's login credentials, fetch relevant event data from the database, and display personalized recommendations. The system will also handle ticket booking and send reminders.

`

Outputs	The dashboard will display a list of recommended events, ticket booking confirmations, and reminders for upcoming events. For example, selecting an event will allow the student to book a ticket, after which a digital ticket will be generated and stored on the platform.

3.2.2 Functional Requirements for Event Organizer Dashboard

Purpose	**The event organizer dashboard allows event managers to create, edit, and manage events, including ticket sales and attendee tracking.**
Inputs	Event organizers will input event details such as event name, date, time, and ticket pricing. They can also manage attendee lists and send out notifications to registered participants.
Processing	The platform will validate and store event details in the database, generate digital tickets, and track ticket sales in real-time. It will also facilitate communication with attendees through notifications and updates.
Outputs	The dashboard will display a summary of event details, ticket sales, and attendee information. For example, selecting an event will show the number of tickets sold and the list of registered attendees.

3.2.3 Functional Requirements for Admin Dashboard

Purpose	**The admin dashboard provides overall control and monitoring capabilities, allowing administrators to manage users, events, and platform settings.**
Inputs	Administrators will input platform settings, user management details, and oversee event approvals and monitoring.
Processing	The platform will process administrative commands, manage user roles, and ensure smooth platform operations by overseeing event creation and user activities.
Outputs	The dashboard will display system status, user activities, and event summaries. For example, selecting a user will show their activity history and permissions.

`

3.3 Non-Functional Requirements

- **Performance:**
  The website must respond to user actions within 2 seconds under normal operating conditions. The system must handle a user base of up to 2000 concurrent users during peak times, especially around major university events and the system should support at least 50 ticket bookings per minute without degradation in performance.

- **Scalability:**
  The database must support the growth in the number of events, users, and bookings over time without performance issues. The system architecture should allow for future integration with other university systems and external event services, as well as expansion to include larger or external events.

- **Security:**
  The platform must use secure user authentication mechanisms, integrating with the university's existing login portal systems like LMS and webkiosk.thapar.edu.
  All sensitive data will be encrypted and different users will have appropriate permissions

- **Reliability:**
  The system must provide clear error messages and handle unexpected conditions properly, ensuring minimal disruption to user activities (e.g., invalid login credentials).

- **Usability:**
  The platform must have an intuitive and user-friendly interface. The UI must follow responsive design principles to ensure compatibility across devices (desktops and smartphones).
- **Backup and Recovery:**
  Regular data backups must be implemented to protect event, user, and ticketing information. The system must have a disaster recovery plan in place to restore services within a few hours in the event of a major failure.

3.4 Hardware Interfaces

- **None specific:** The platform will be web-based and does not require specific hardware interfaces.

3.5 Quality Attributes

- **User Experience**: The platform will prioritize quick load times and smooth navigation across all supported devices, ensuring an enjoyable user experience.
- **Error Handling**: The system will be robust, tolerating a wide variety of inputs, including incorrect or unexpected user actions, with appropriate error messages and recovery options.

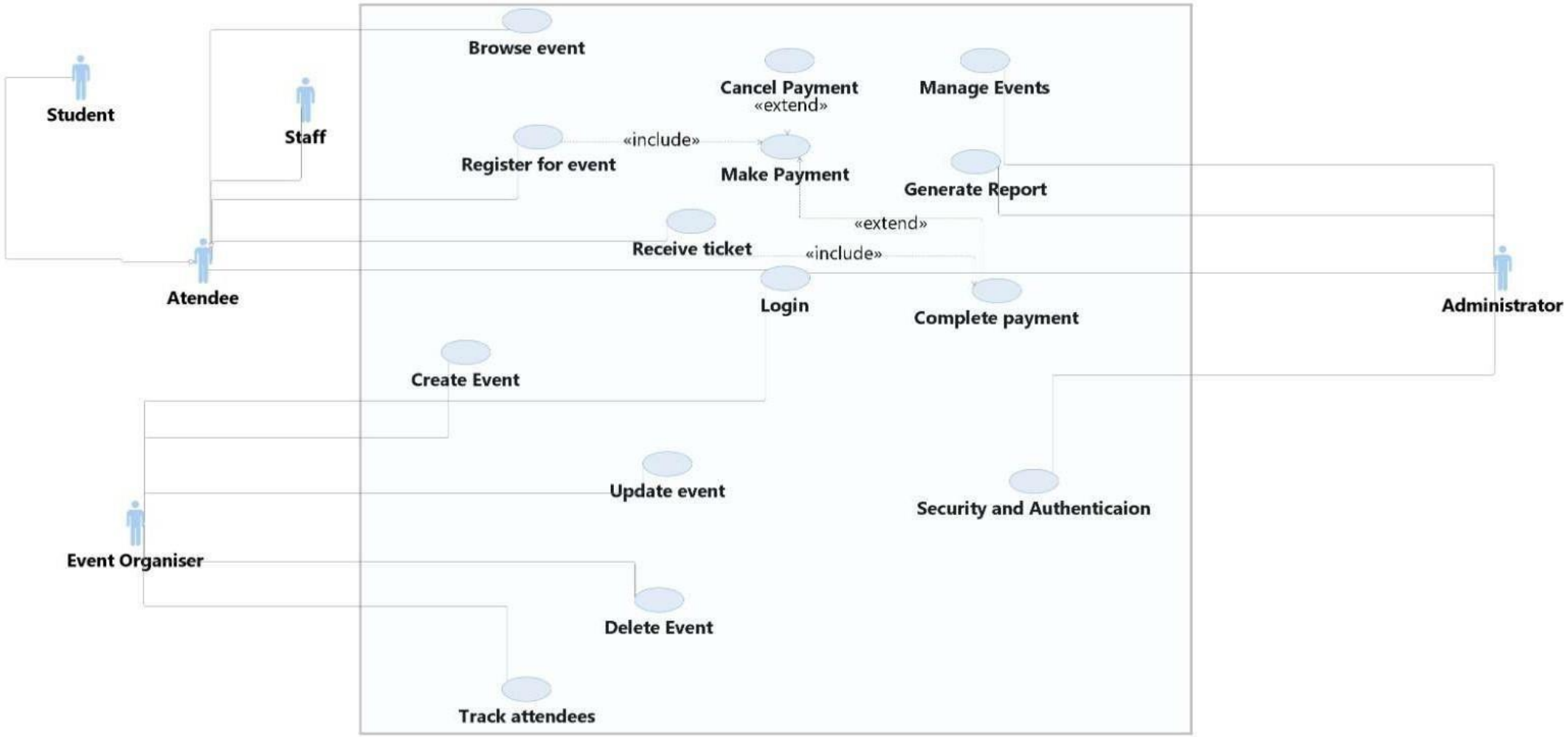
`

- ## 3.6 Other Requirements

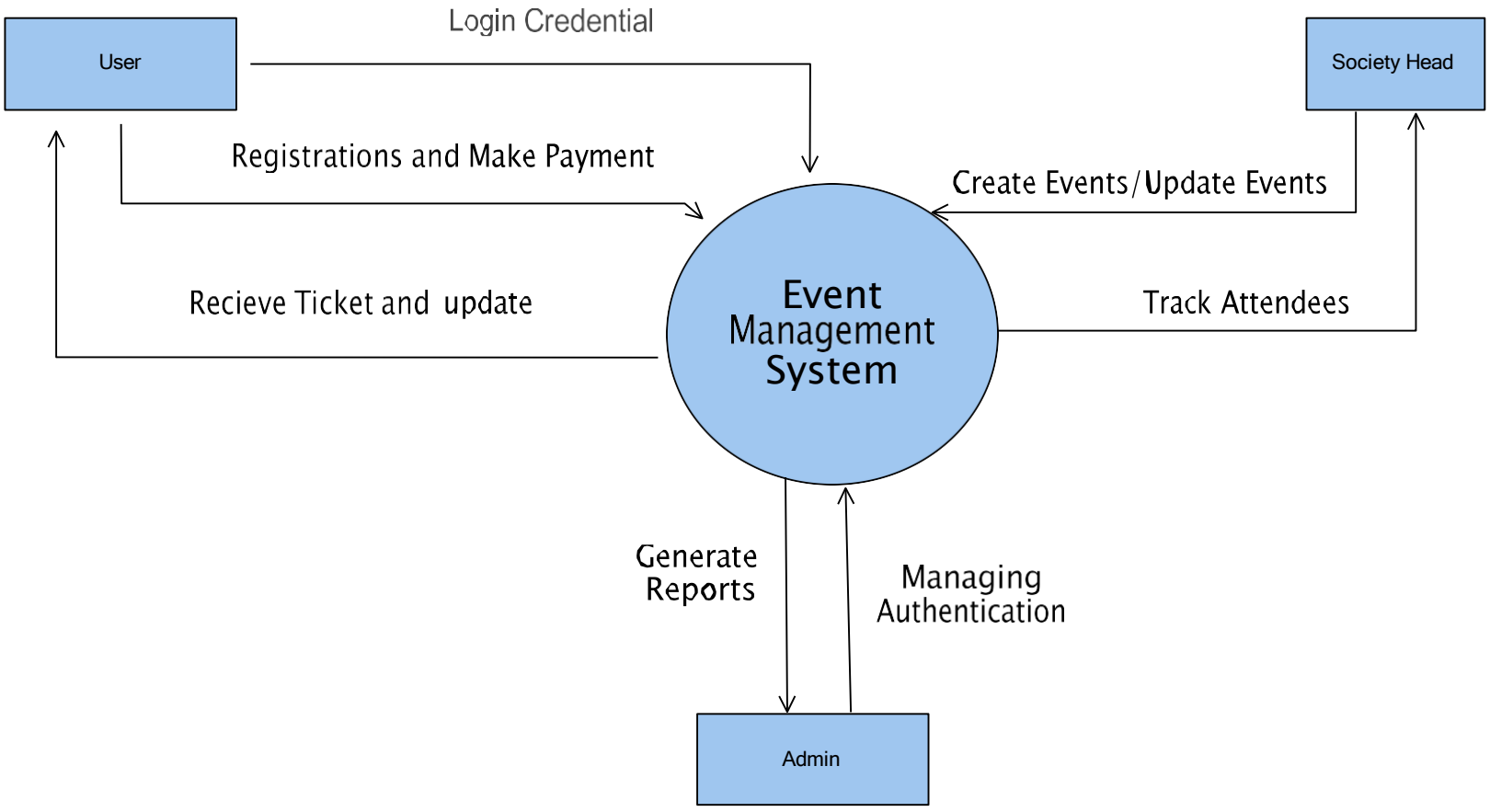
At this stage, no additional requirements have been identified. However, future updates may include features like sound notifications and advanced analytics for event organizers.

`

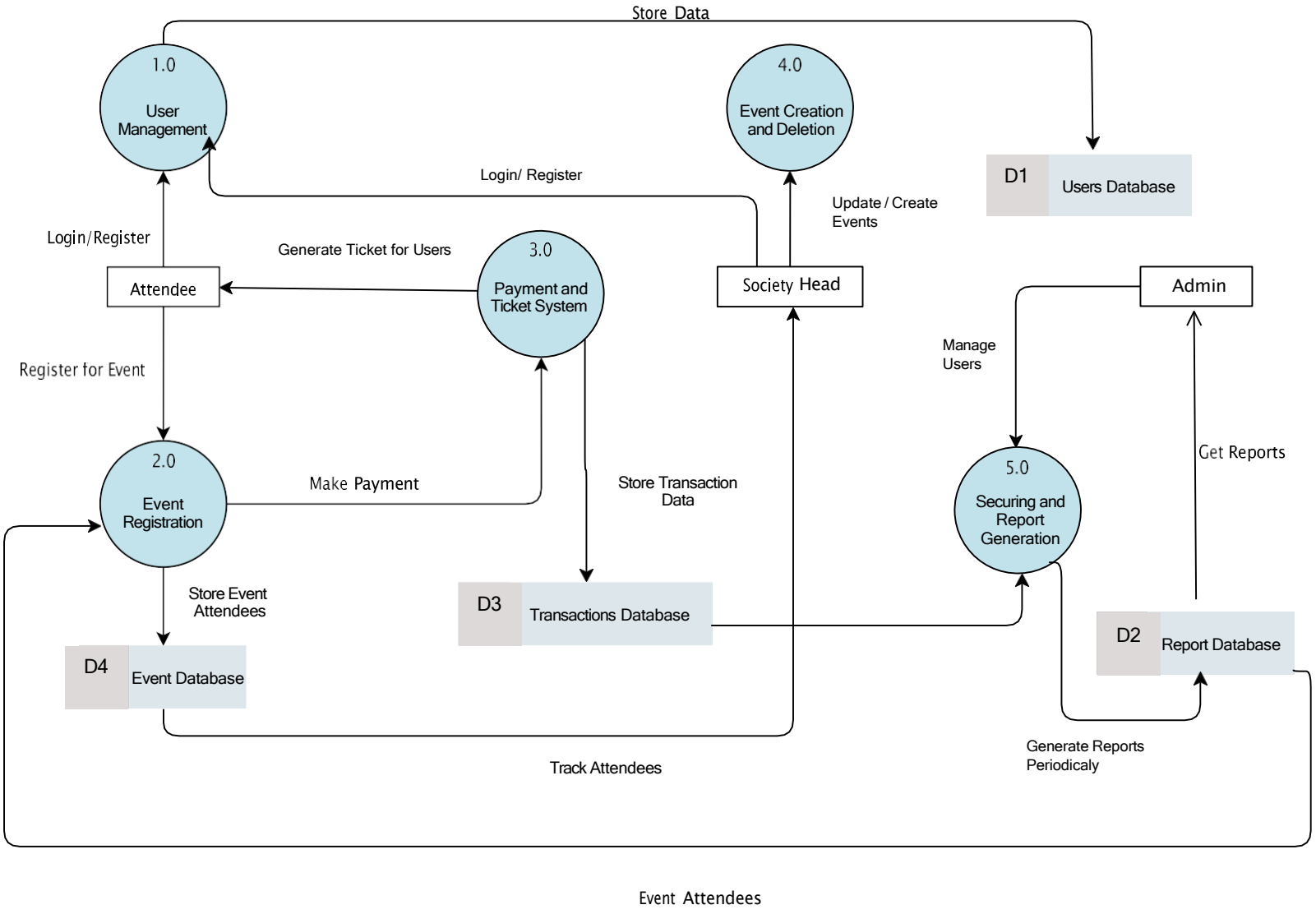
Use-Case Diagram



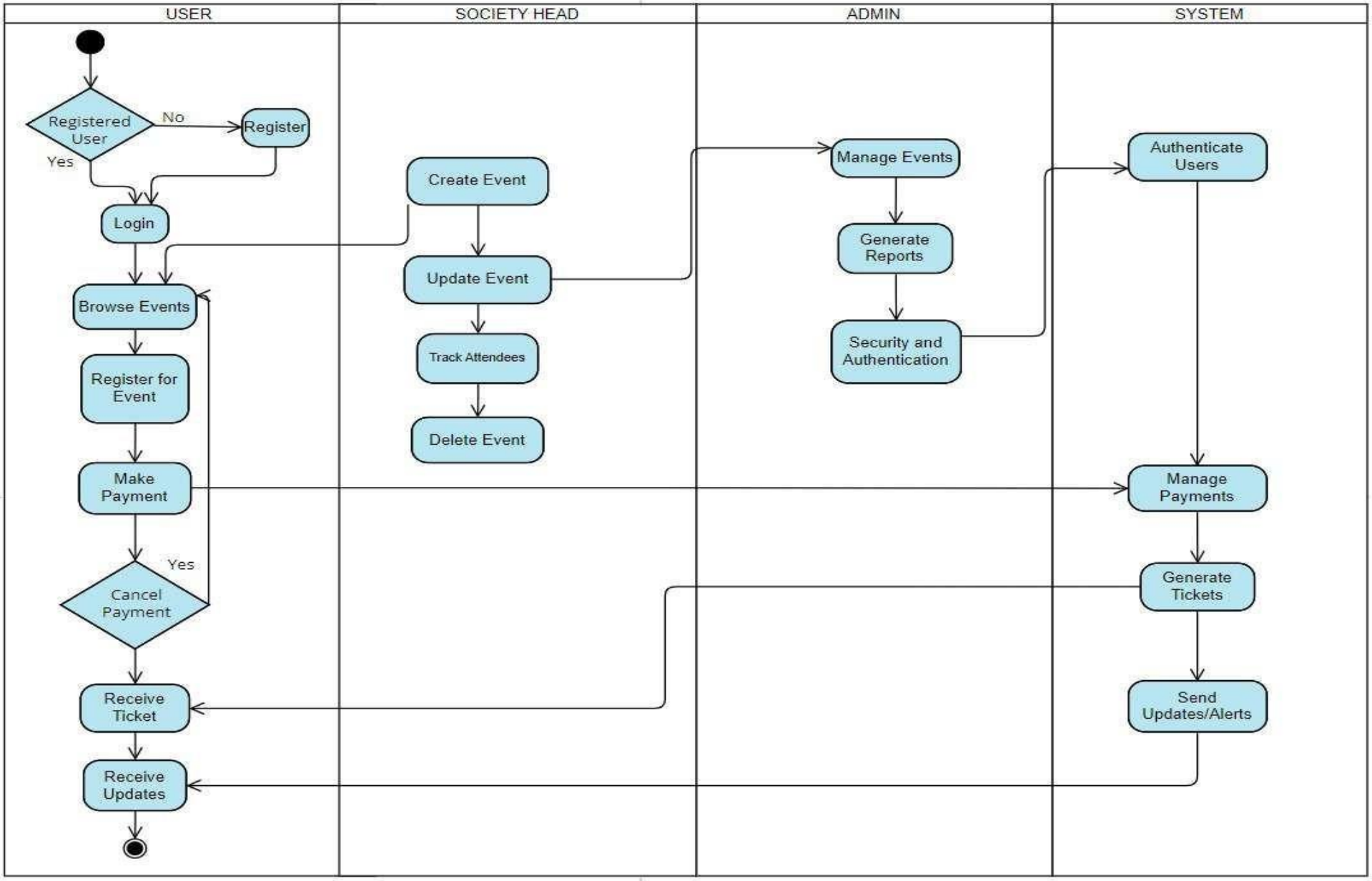
DFD Level-0


User
Login Credential
Society Head
Registrations and Make Payment
Create Events/Update Events
Recieve Ticket and update
Event Management System
Track Attendees
Generate Reports
Managing Authentication
Admin

DFD Level-1

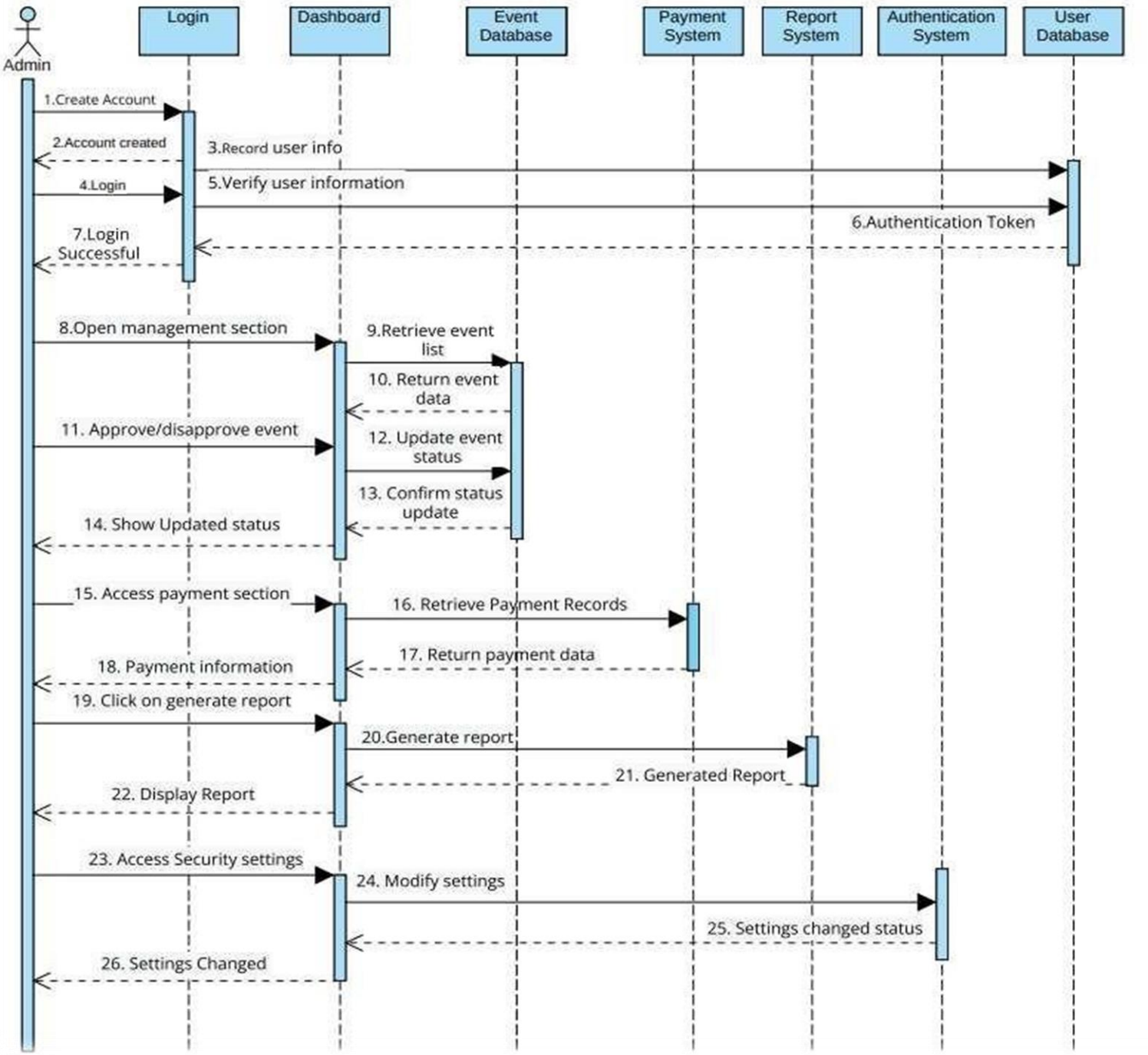

Store Data
1.0
User Management
4.0
Event Creation and Deletion
D1
Users Database
Login/ Register
Update / Create Events
Login/Register
Generate Ticket for Users
3.0
Payment and Ticket System
Attendee
Society Head
Admin
Manage Users
Register for Event
2.0
Event Registration
Make Payment
Store Transaction Data
5.0
Securing and Report Generation
Get Reports
Store Event Attendees
D3
Transactions Database
D2
Report Database
D4
Event Database
Track Attendees
Generate Reports Periodicaly
Event Attendees

Activity Diagram



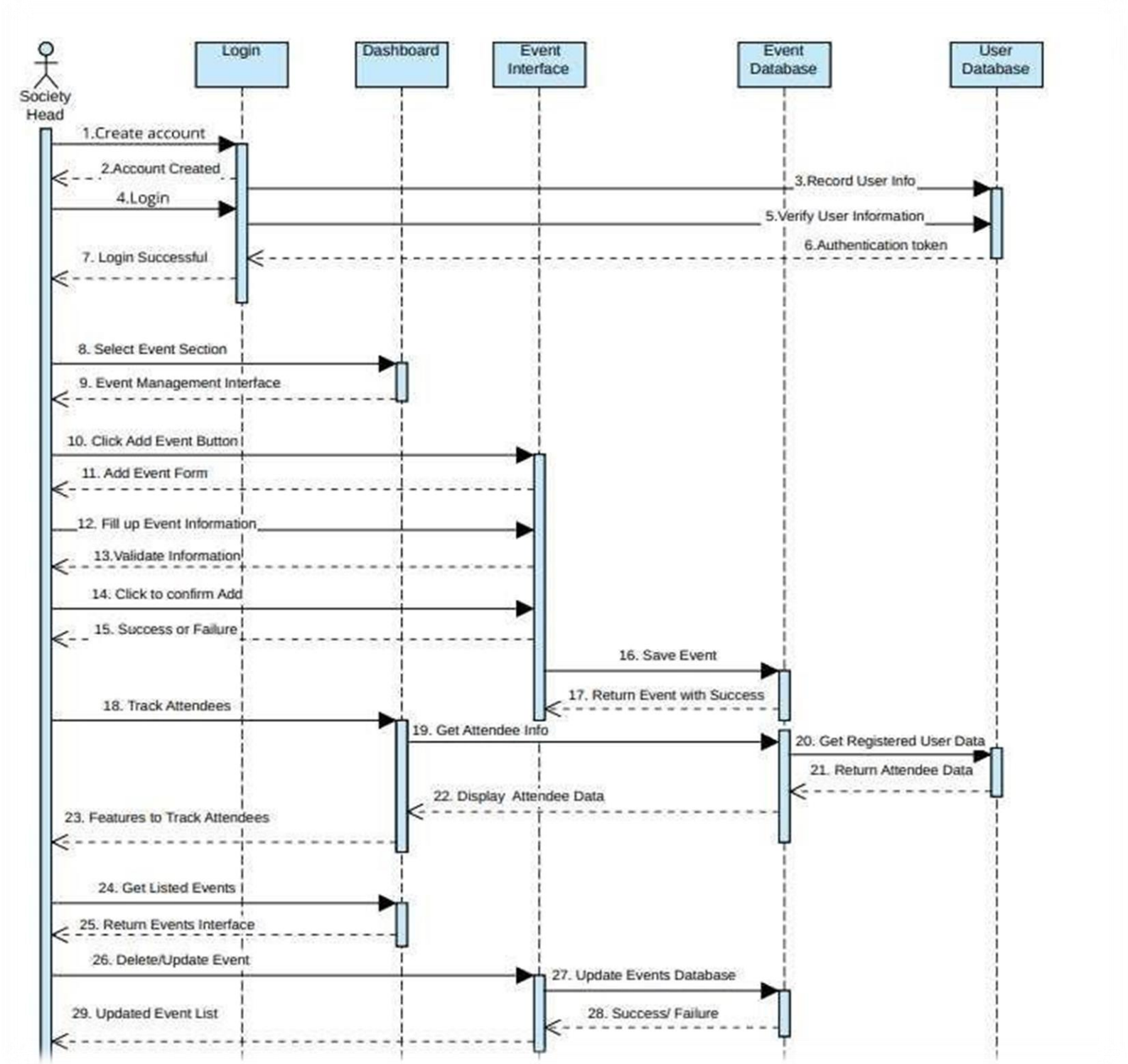
`

Sequence-Diagram (Admin)



`

Sequence-Diagram (Society Head)



Sequence-Diagram (User)