

---

# **Software Requirements Specification**

**for**

# **Event Management System**

**Version 3.0**

**Prepared by**  
**Ganesh P Nischay 16IT220**  
**Sai Kumar 16IT241**  
**Supreeth G 16IT246**

**National Institute of Technology Karnataka, Surathkal**

**21<sup>st</sup> February, 2019**

# Table of Contents

<b>Table of Contents.....</b>	<b>2</b>
<b>Revision History.....</b>	<b>3</b>
<b>1. Introduction.....</b>	<b>4</b>
1.1 Purpose.....	4
1.2 Document Conventions.....	4
1.3 Intended Audience and Reading Suggestions.....	4
1.4 Product Scope.....	5
1.5 References.....	5
<b>2.Overall Description.....</b>	<b>5</b>
2.1 Product Perspective.....	5
2.2 Product Functions.....	6
2.3 User Classes and Characteristics.....	6
2.4 Operating Environment.....	6
2.5 Design and Implementation Constraints.....	6
2.6 User Documentation.....	6
2.7 Assumptions and Dependencies.....	6
<b>3.External Interface Requirements.....</b>	<b>7</b>
3.1 User Interfaces.....	7
3.2 Hardware Interfaces.....	7
3.3 Software Interfaces.....	7
3.4 Communications Interfaces.....	7
<b>4.System Features.....</b>	<b>8</b>
4.1 Authentication: Login & Registration.....	8
4.2 Organize /Add new event.....	8
4.3 Buy ticket/Use invitation for an event.....	9
4.4 Modify details of upcoming event.....	10
4.5 Display past events.....	11
4.6 Display user profile.....	11
4.7 Add money to wallet.....	12
<b>5. Other Nonfunctional Requirements.....</b>	<b>12</b>
5.1 Performance Requirements.....	12
5.2 Safety Requirements.....	12
5.3 Security Requirements.....	13
5.4 Software Quality Attributes.....	13
5.5 Business Rules.....	13
<b>6. Other Requirements.....</b>	<b>13</b>
<b>Appendix A: Glossary.....</b>	<b>13</b>
<b>Appendix B: Analysis Models.....</b>	<b>14</b>
<b>Appendix C: To Be Determined List.....</b>	<b>28</b>

## Revision History

Name	Date	Reason for Changes	Version
Ganesh, Sai, Supreeth	18/01/2019	Added project description.	1.0
Sai Kumar, Supreeth	25/01/2019	Added System functional & non-functional requirements, and use-case diagram	2.0
Ganesh	08/02/2019	Added sequence, ER, State diagrams.	3.0

# 1. Introduction

This section gives a scope description and overview of everything included in this SRS document. Also, the purpose of the software being developed as well as the intended audience is provided.

## 1.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the “Event Management System” software. This is the first release of the third version of the software. This document will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended as a reference for developing the first version of the system for the development team.

## 1.2 Document Conventions

The document uses the following conventions:

- DB: Database
- SRS: Software Requirements Specification
- Event-manager: Refers to the user who organizes/adds a new event. He/she is the manager of the event.
- Event-attendee: Refers to the user who wishes to attend an event either by buying a ticket or using an invitation.
- Home-page: The main web page of the application where all upcoming events are shown.

These are the conventions that were used for editing the document:

- The document follows the IEEE Software Requirements Specification format.
- This document uses the Calibri style with size 12.

## 1.3 Intended Audience and Reading Suggestions

This document is primarily intended for the project leader and the product development team. It is also intended for the testers and the documentation writers.

The remainder of this document includes three chapters :

- The second chapter provides an overview of the system functionality and system interaction with other systems. This chapter also introduces different types of stakeholders and their interaction with the system. Further, the chapter also mentions the system constraints and assumptions about the product.

- The third chapter provides the requirements specification in detailed terms and a description of the different system interfaces. Different specification techniques are used in order to specify the requirements more precisely for different audiences.
- The fourth chapter deals with the prioritization of the requirements. It includes a motivation for the chosen prioritization methods and discusses why other alternatives were not chosen.

## **1.4 Product Scope**

The Event Management System is a software application which is intended for organizing and managing events using a web interface. The users can be either event-managers or event-attendees or both. The event-managers can add new events specifying the location, date and time. Event attendees can register for these events either by buying tickets or using an invitation which is given by the event-manager. The main goal of the system is to ease the process of organizing events and provide a means for buying tickets to attend these events.

## **1.5 References**

[1] IEEE Software Engineering Standards Committee, "IEEE Std 830-1998, IEEE Recommended Practice for Software Requirements Specifications", October 20, 1998.

# **2. Overall Description**

This section will give an overview of the whole system. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionality of it. It will also describe what type of stakeholders will use the system and what functionality is available for each type. At last, the constraints and assumptions for the system will be presented.

## **2.1 Product Perspective**

The software product being developed is a new, self-contained product being developed for the first time. The product is entirely a web based application. The application will be used to display events and allow users to register for upcoming events. The data regarding events must be provided by the event-managers. Event-managers are the organizers of the event. Since, this is a data-centric product, it will need somewhere to store the data. For that, a database will be used. The web application can modify the database based on what the users require.

## 2.2 Product Functions

- The users of the product acting as event-managers will be able to register their events in the system.
- The product also helps other users to attend these events by helping them to register for these events either by buying tickets or by using an invitation card.
- Users can also search for events based on the event name, event location or date.
- The event-managers can postpone or prepone their events if necessary and a mechanism will be provided to inform all event-attendees.
- In order to add a new event or buy a ticket, the user must login to the system.

## 2.3 User Classes and Characteristics

The major user classes who will use the system are the event-managers and the event-attendees. A user can be both an event-manager and an event-attendee, but a user can only be either be an event manager or an event-attendee for a particular event. The event-manager is the user organizing his/her event. The event-manager registers his/her event on the website. While registering the event, the event-manager must fill the details of the event like the location, date and time. The event- managers can also edit the details of their events at a later stage. The event-attendees are the users who either buy tickets to attend events or use an invitation card to attend the event.

## 2.4 Operating Environment

The software being a web application can run on any operating system. The suggested web browser for running the application is Google Chrome. Any browser can be used provided it supports JavaScript functionalities. The hardware platform can be any desktop, mobile or laptop device. Apart from Google Chrome, Microsoft Edge, IE8 and Mozilla Firefox are also suggested.

## 2.5 Design and Implementation Constraints

The web application will be constrained by the capacity of the application framework being used for development and the efficiency also depends on the integration of the DB into the application. The DB may in some cases be forced to queue incoming requests and therefore increase the time it takes to fetch data.

## 2.6 User Documentation

TBD

## **2.7 Assumptions and Dependencies**

- The users must be logged in to add a new event or buy tickets for a new event.
- Some events can be attended only by those who have received an invitation.
- The payment for the tickets is assumed to be completed only from an online wallet provided to each user when he/she registers an account in the system. A unique identification PIN is associated with each wallet which the user provides during registration. This PIN is necessary for successful completion of the transaction.

## **3. External Interface Requirements**

### **3.1 User Interfaces**

TBD

### **3.2 Hardware Interfaces**

TBD

### **3.3 Software Interfaces**

- Django - a web framework built using python will be used for developing the web application.
- SQLite DB will be used in this application, as it is a robust and easy to use database and can be easily integrated with the framework.
- The prescribed browser to run the application is Google Chrome. Other browsers like Microsoft Edge, Mozilla Firefox, IE8 can also be used.
- The application can run on any operating system.

### **3.4 Communications Interfaces**

TBD

## 4. System Features

This section describes the various features and functional requirements of the Event Management System application.

### 4.1 Authentication: Login & Registration

#### 4.1.1 Description and Priority

The event management system includes login and registration pages for the users of the system. This is required for user authentication. The different users will have to enter their login credentials in order to add a new event or to attend an upcoming event. Once the users have successfully logged in, they will be redirected to the desired page. This feature has a high priority as it prevents unauthorized access into the system.

#### 4.1.2 Response Sequences

If the user is using the system for the first time, he must register an account in order to use the system. During registration, the user must fill all the necessary fields which will be displayed in the registration page. The user will have to provide an email-id and set a password for his/her account. If registration is unsuccessful, an appropriate message will be displayed to the user.

Registered users will have to login using their credentials whenever they want to use the system. Upon successful authentication, the users will be redirected to the home page where they can add a new event as an event-manager or book a ticket for an upcoming event as an event-attendee. A message will be displayed to the user if the login was unsuccessful.

#### 4.1.3 Functional Requirements

REQ-1: All users of the system must have an account

REQ-2: All the necessary fields must be filled in the registration page

REQ-3: Every user of the system must have unique login credentials

REQ-4: Authentication is necessary for managing events

### 4.2 Organize/Add new event

#### 4.2.1 Description and Priority

After successfully authenticating, the user can organize a new event as the manager of the event. The option for organizing a new event is available as a button in the home-page of the web application. The home-page contains a list of all upcoming events. The new event once added will appear in the home-page and other users can buy tickets for these events.



This feature is of high priority as it is the core of the application: without new events there will be no events and hence no event management.

#### **4.2.2 Response Sequences**

The user must click on the 'Add Event' button present in the home page in order to add a new event as the event-manager. Upon clicking this button, a form will appear and this form must be filled with the details of the event such as the event name, location, date, time, ticket fare, etc. If the new event being created already exists, a corresponding message will be displayed to the user. Upon successful submission of the form, the event will be created and displayed in the home-page of the application and other users can buy tickets to attend these events.

#### **4.2.3 Functional Requirements**

REQ-1: The user must have logged in to add a new event

REQ-2: All the necessary fields must be filled in the event description form for adding a new event.

REQ-3: No two events organized by the same user must be same. In other words duplication is not allowed and a suitable message will be displayed.

### **4.3 Buy ticket/ Use invitation for an event**

#### **4.3.1 Description and Priority**

The users can attend upcoming events either by buying tickets for the event or by using an invitation card. The upcoming events are displayed in the home-page of the web-application. The event-manager can send invitation cards to specific users, who the event-manager wishes to invite. These users can use the invitation card to attend the event. If it is necessary to buy tickets for the event, the users can do so as well. This feature of buying tickets / using invitation to attend an event is of high priority.

#### **4.3.2 Response Sequences**

The user must select an event from the home-page which he/she wishes to attend. Upon clicking an event, the user will be redirected to the event description page where the user can view the details regarding the event like location, date, time, ticket fare, etc. If the user wishes to attend the event he can use an invitation card to inform that he is attending the event by clicking the 'use invitation card' button displayed in the event description page. If the user has to buy a ticket, he can do so by clicking on the 'buy ticket button' also displayed in the event description page. If the user wants to buy a ticket, he must have sufficient money in his wallet in order to buy the ticket. Upon clicking the buy ticket button, a

confirmation message will be displayed and the amount equal to the ticket fare will be deducted from the users' wallet.

#### **4.3.3 Functional Requirements**

REQ-1: The user must be logged in to buy tickets.

REQ-2: The user must have an invitation id to use an invitation card.

REQ-3: When a user selects an event, all the details of the event such as the event name, location, date, time, ticket fare must be fetched from the database and displayed to the user.

REQ-4: The user must have sufficient amount in his/her wallet to buy tickets. If the amount is less than the cost of the ticket, a corresponding error message will be displayed to the user and the user will be redirected to the add money page.

### **4.4 Modify details of upcoming events**

#### **4.4.1 Description and Priority**

The event-manager can modify details of his/her event by selecting them from the list present in the home-page of the web application. The event-manager may want to change the name, location, time or may want to postpone or prepone the event. The event-attendees who have bought tickets for the event will be notified about the changes done by the event-manager. This feature is of medium priority as the event-managers can be asked not to change the event details once they have been set upon creation of the event.

#### **4.4.2 Response Sequences**

The event-manager must select the event from the list present in the home page which he/she wishes to modify. Upon clicking the event, the user will be redirected to the event description page. Here, the user must click the modify details button and make the necessary changes in the form which appears. Upon clicking the submit button, the necessary changes will be saved to the database and displayed in the home page.

#### **4.4.3 Functional Requirements**

REQ-1: The event-manager must be logged in to modify the details of the event.

REQ-2: Only the event-manager must be able to modify the details of the event. REQ-3: Upon successful modification of the event details, the changes must also be made in the database.

## **4.5 Display past events**

### **4.5.1 Description and Priority**

The users of the system can view past events that were organized with the help of the event management system. The option to view past events is present in the home-page of the web application in the form a button. This feature has low priority as it has no major importance other than storing old data.

### **4.5.2 Response Sequences**

The users must click on the view past events button present in the home-page of the application. Upon clicking this button the user will be redirected to a page where a list of all old events will be displayed. The user can click on any of these events to view the event description.

### **4.5.3 Functional Requirements**

REQ-1: User must be logged in.

REQ-2: Data of old events must be stored in a database and retrieved when the user clicks on the view old events button.

## **4.6 Display user profile**

### **4.6.1 Description and Priority**

The users can view their profile which contains basic information of the user such as name, events organized, past events attended and upcoming events attending. The user must be logged in to view the profile. This feature is of low priority as it has no major function with respect to event management.

### **4.6.2 Response Sequences**

The user after logging in can press on the view profile button present in the home-page of the web application. Upon clicking the button the user will be redirected to the user profile page.

### **4.6.3 Functional Requirements**

REQ-1: The user must be logged in.

REQ-2: The data about the user must be retrieved from the database in order to display it on the web page.

## **4.7 Add money to wallet**

### **4.7.1 Description and Priority**

The users can add money to their wallet by using this feature. This feature has high priority as it is the only way they can buy tickets for attending events.

### **4.7.2 Response Sequences**

The user must select the 'add money' button present in the home-page of the web application. Upon clicking this button the user will be redirected to a page where the user must enter the money to be added to the wallet along with the unique PIN provided by the user during account registration. Upon success, the entered amount will be added to the wallet. If the PIN is not correct, an appropriate message will be displayed to the user.

### **4.7.3 Functional Requirements**

REQ-1: The user must be logged in.

REQ-2: The user must enter the right PIN in order to successfully add money to the wallet. The PIN is checked with data found in the database.

## **5. Other Nonfunctional Requirements**

### **5.1 Performance Requirements**

The event management system requires a good browser which can render the pages with good speed. The application also requires a good internet connection. Browsers such as Google Chrome, Mozilla Firefox, IE8 or Microsoft Edge are recommended.

The performance of the system depends on various factors such as the network and server traffic. Server load can dramatically increase with more and more users using the application, thereby increasing the number of database queries.

### **5.2 Safety Requirements**

The user must make sure that he/she has set a strong password in order to prevent his/her account from getting compromised.

### 5.3 Security Requirements

- To prevent unauthorized access to the application and in order to prevent false data manipulation, only users who have registered can add new events.
- Proper form validation must be done in order to prevent security attacks and prevent erroneous modifications to the database.

### 5.4 Software Quality Attributes

- Correctness : The information displayed about the events must be correct.
- Maintainability: The web application should be easy to extend and maintain.
- Portability: The website should be accessible on all kinds of devices.
- Usability: Interface should be user friendly with respect to the media device through which it is being accessed.

### 5.5 Business Rules

- Event-manager : Add/ Organize new event, Modify event details.
- Event-attendees: Buy ticket / use an invitation.

Apart from the above specific functions, all users can login and need to perform registration before using the application. All users can add money to their wallet. They can view their profile and also view all past events managed using the application.

## 6. Other Requirements

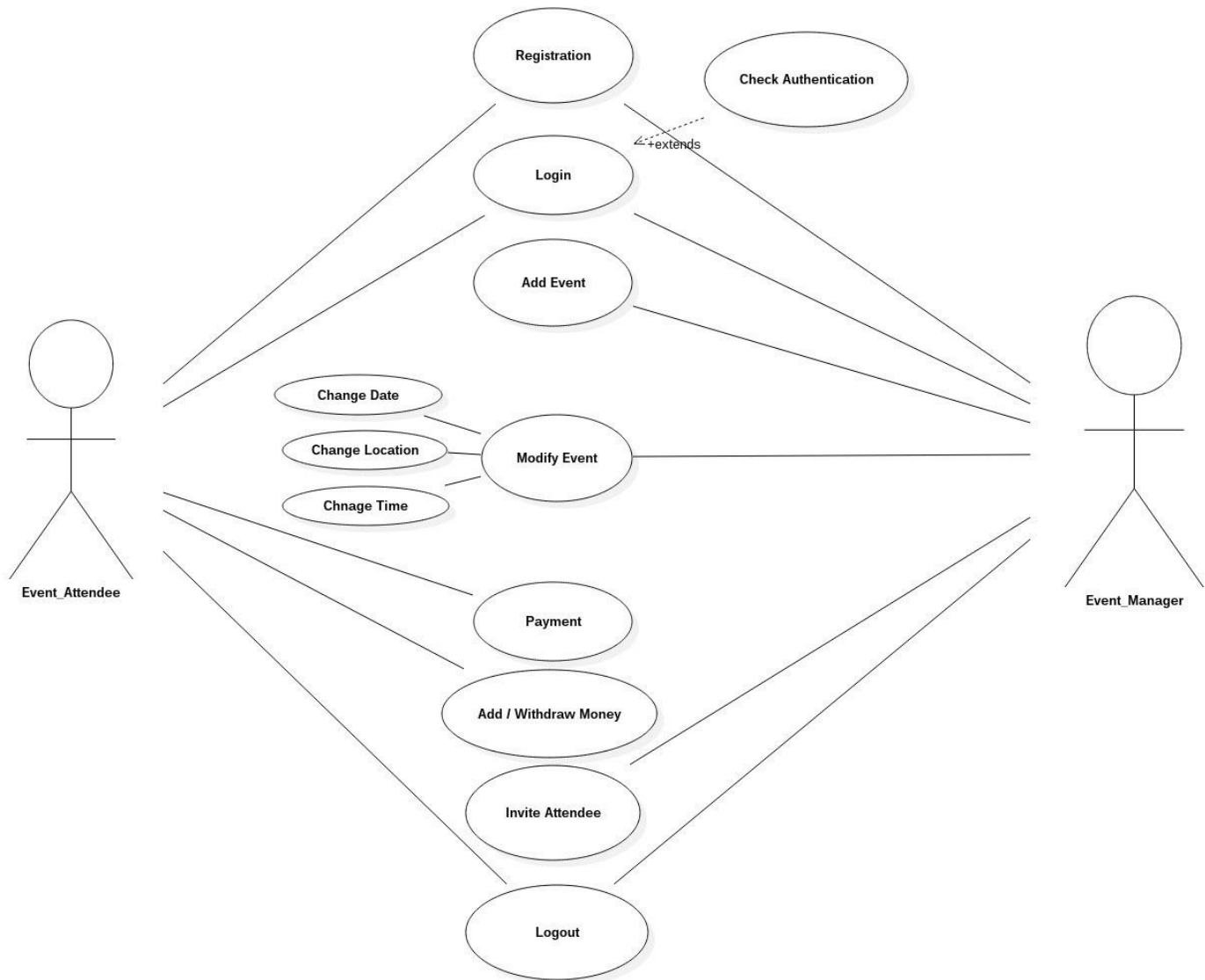
TBD

## Appendix A: Glossary

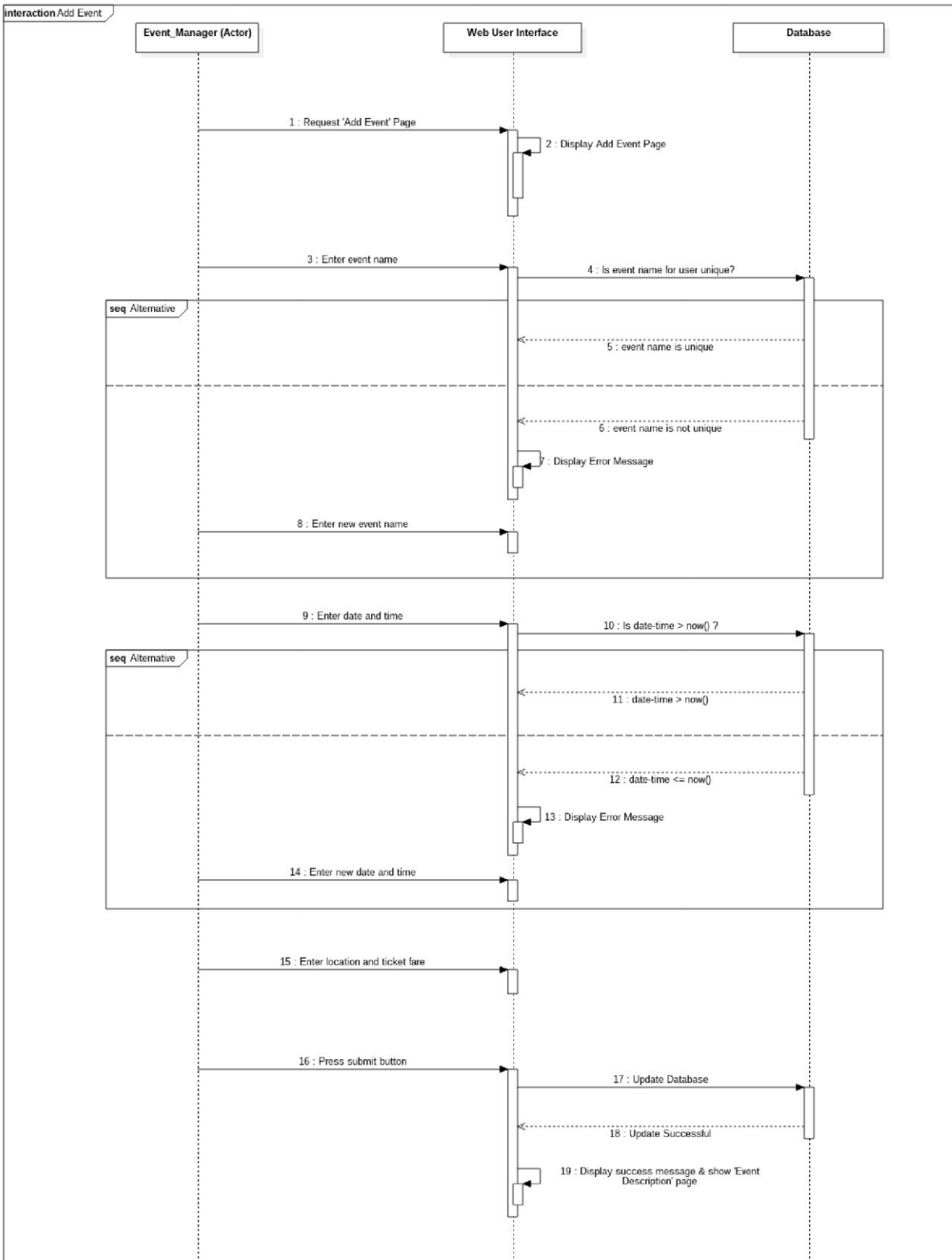
- SRS : Software Requirements Specification
- DB : Database
- IE8 : Internet Explorer 8
- TBD : To be determined

## Appendix B: Analysis Models

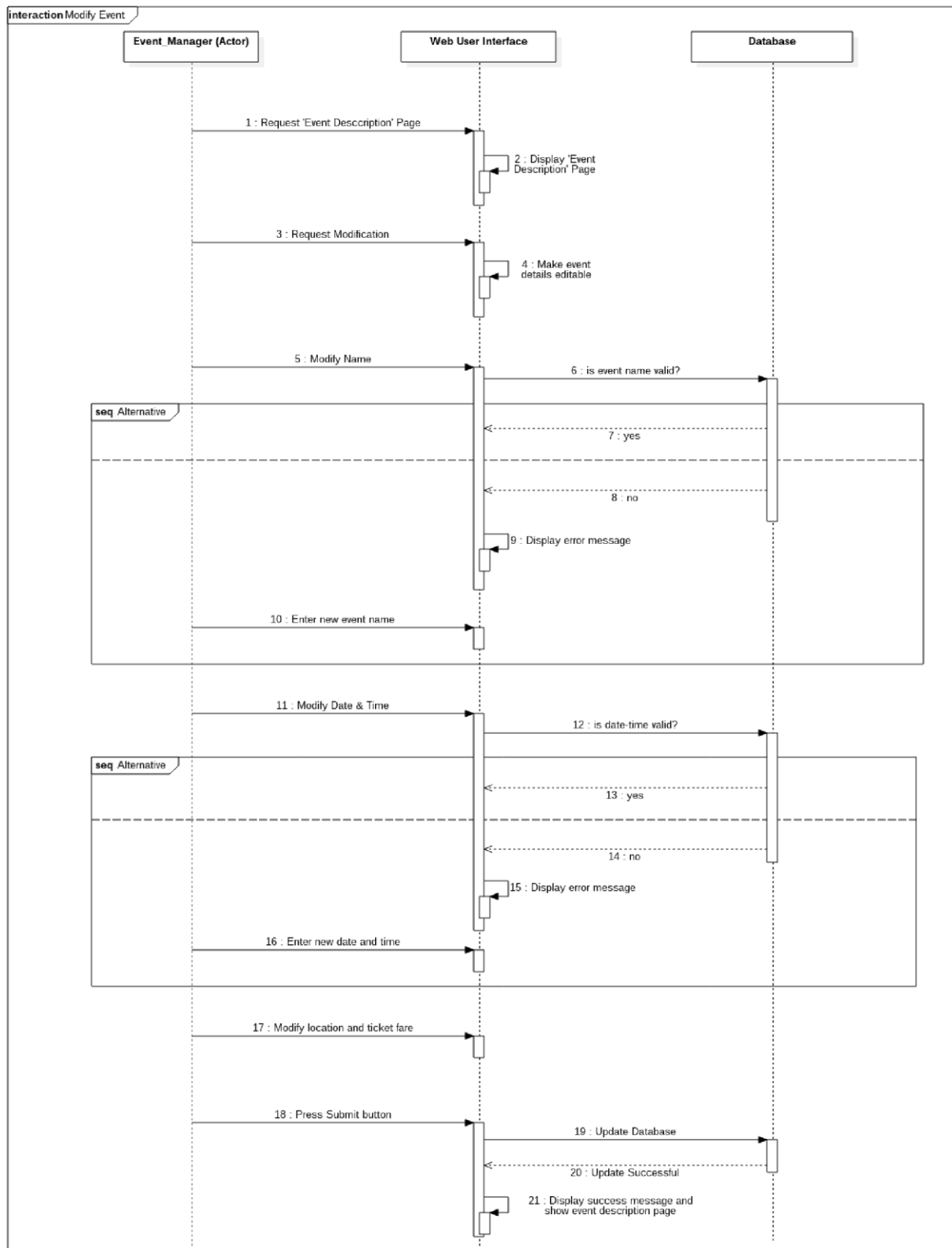
### 1. Use Case Diagram



## 2. Sequence Diagram I. Add Event

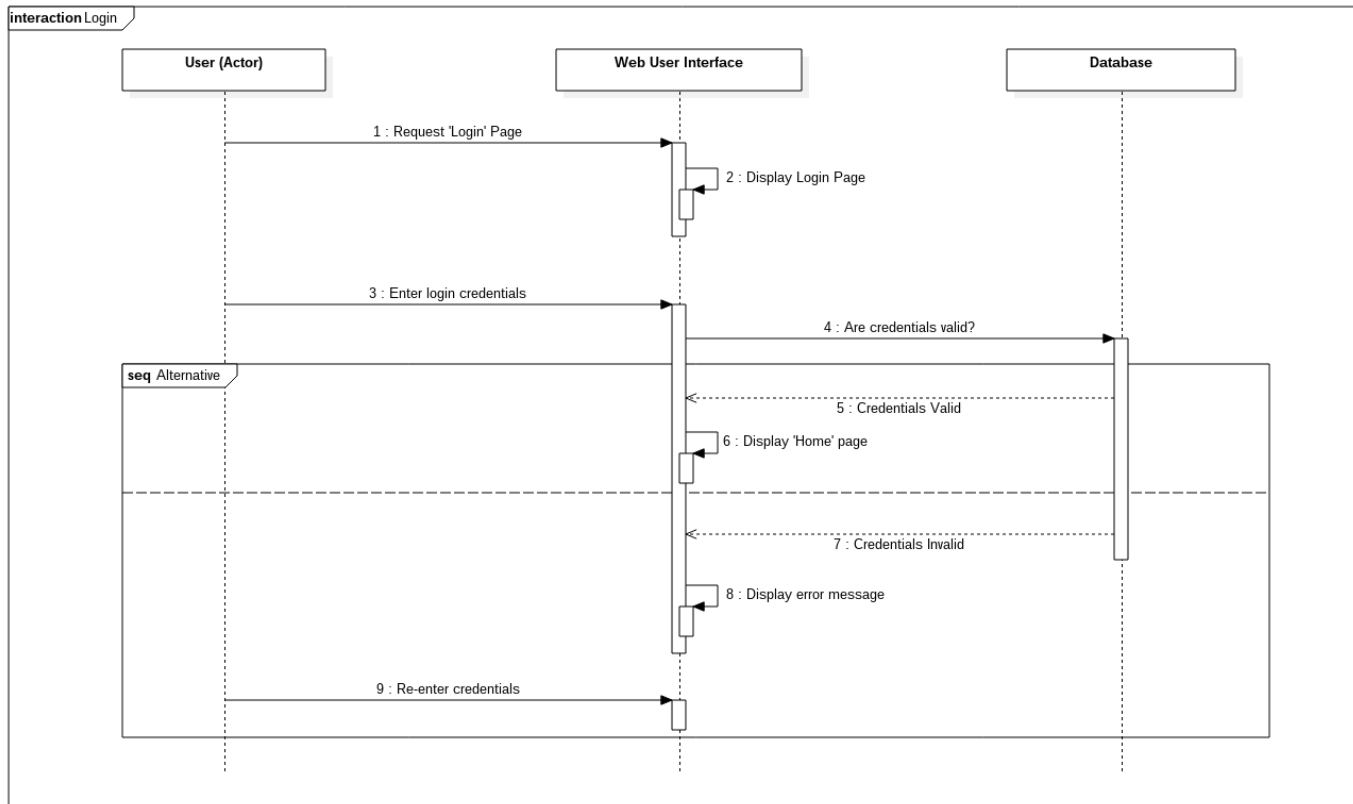


## II. Modify Event

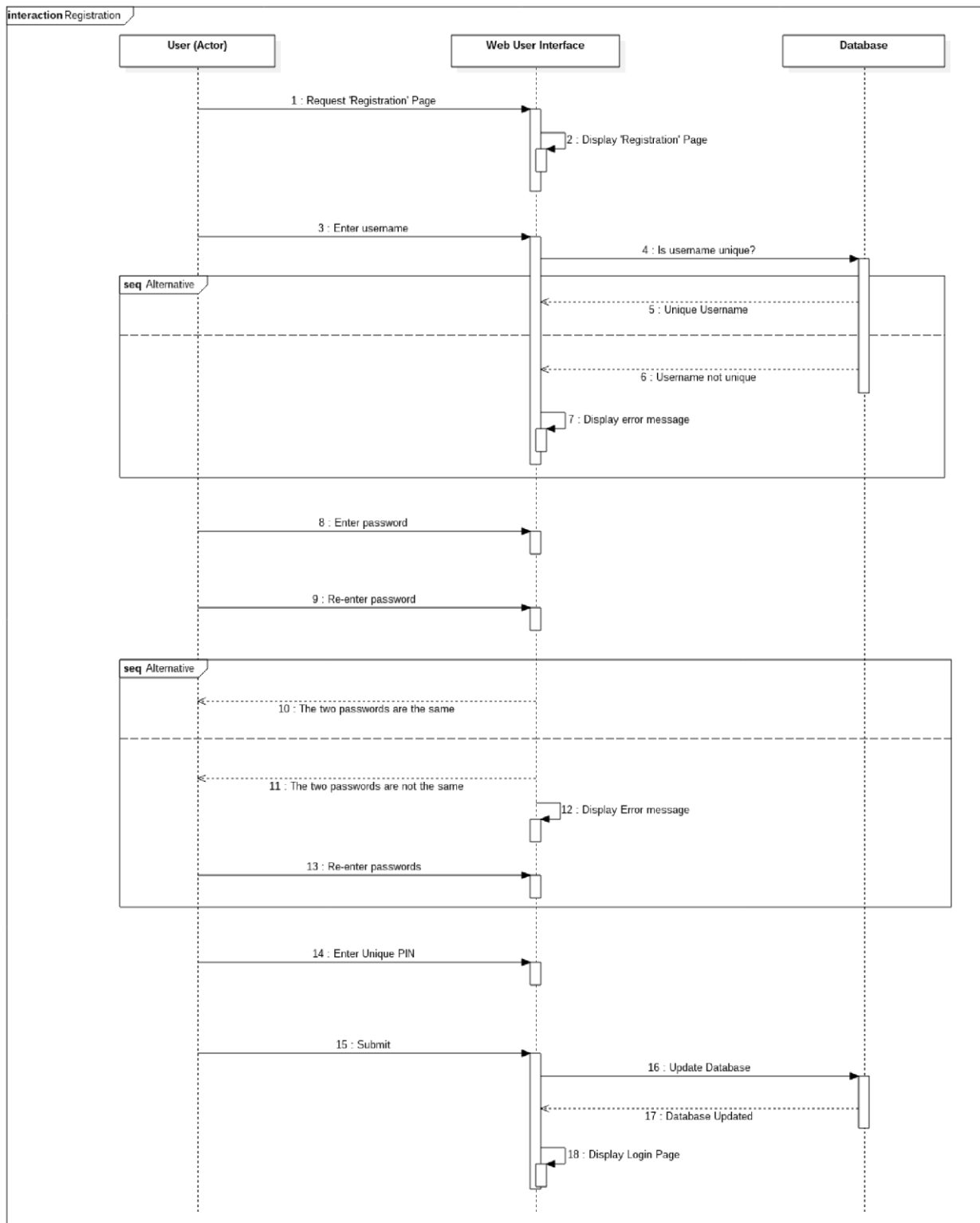




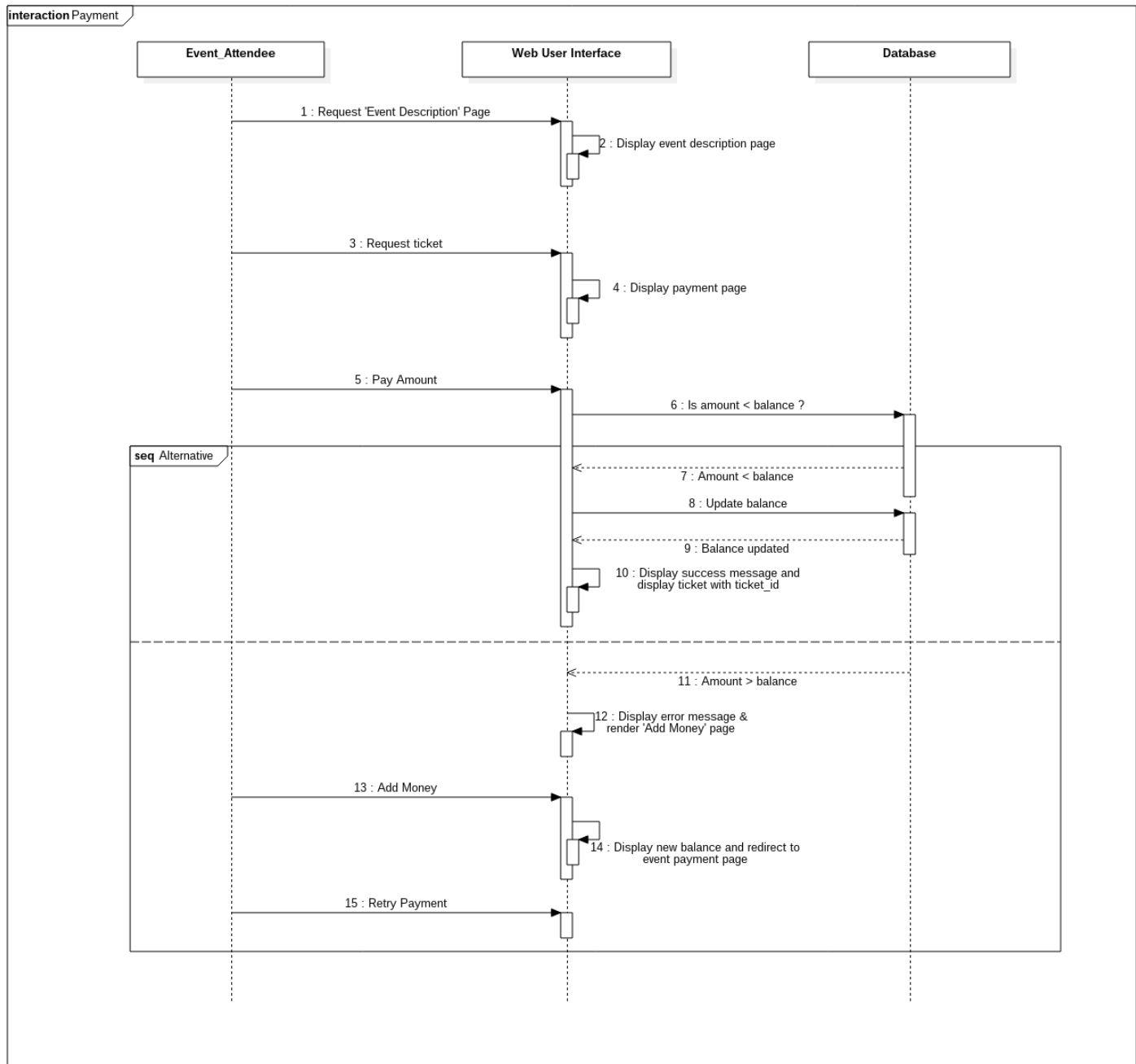
### III. Login



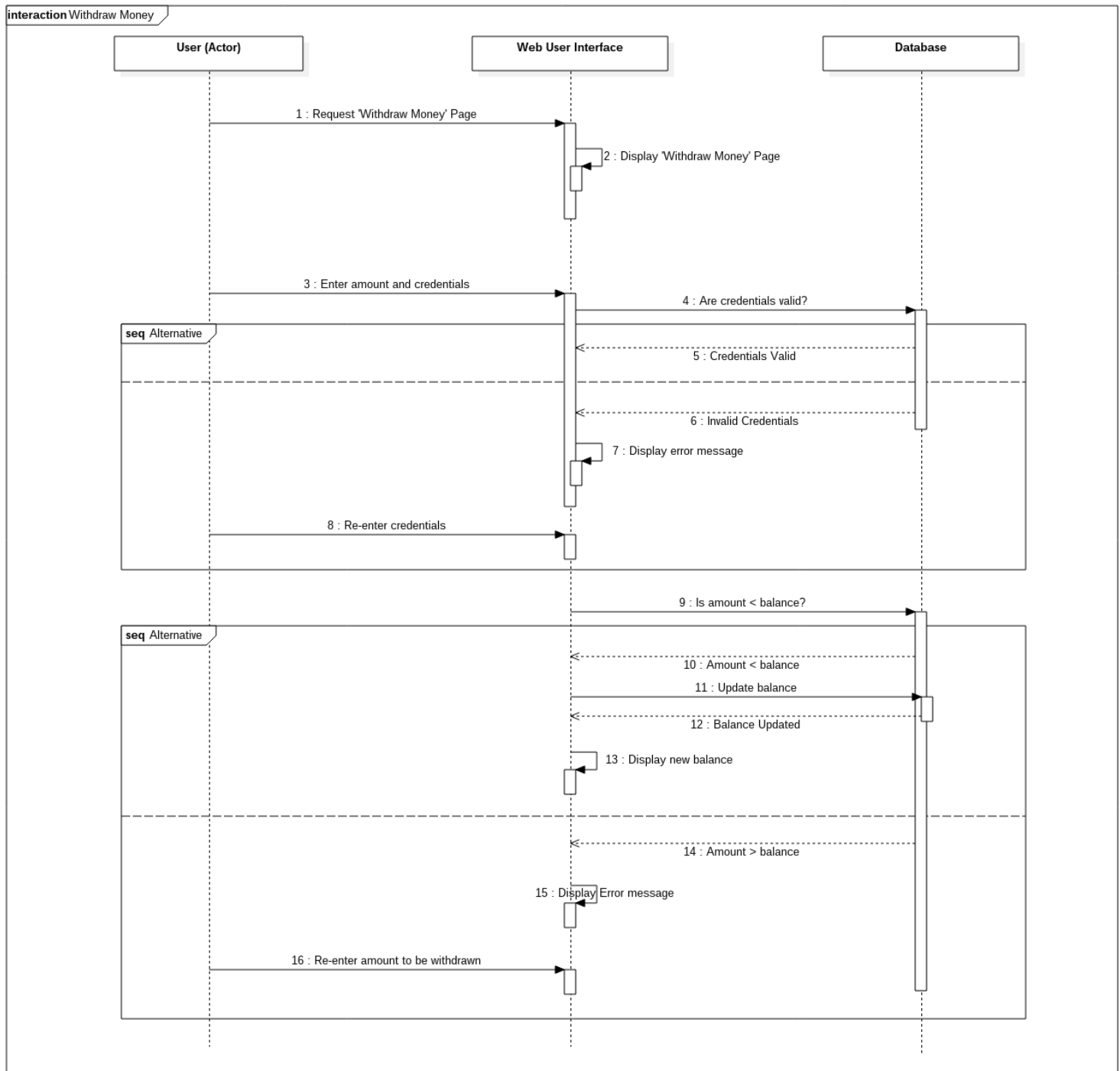
## IV. Register



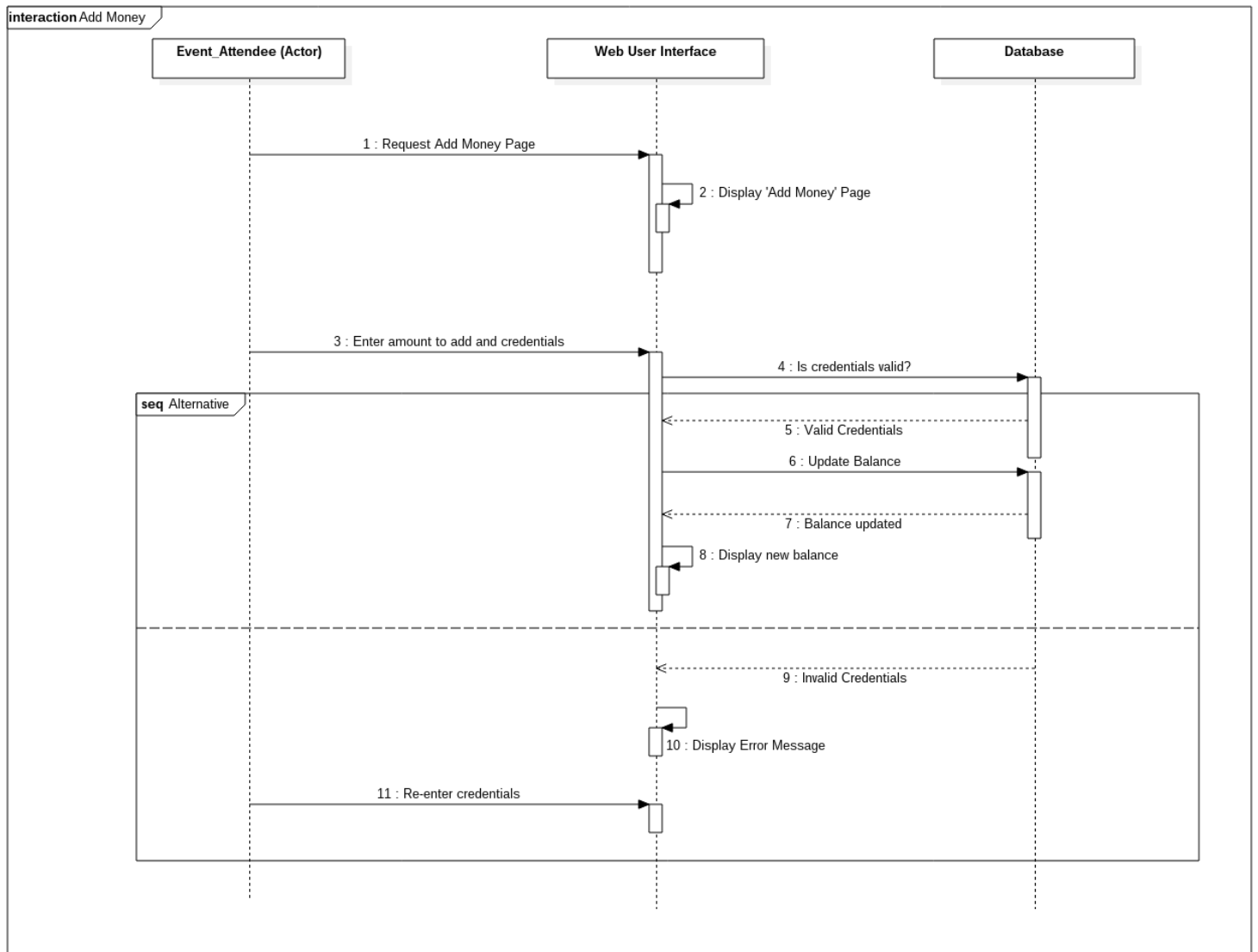
## V. Payment



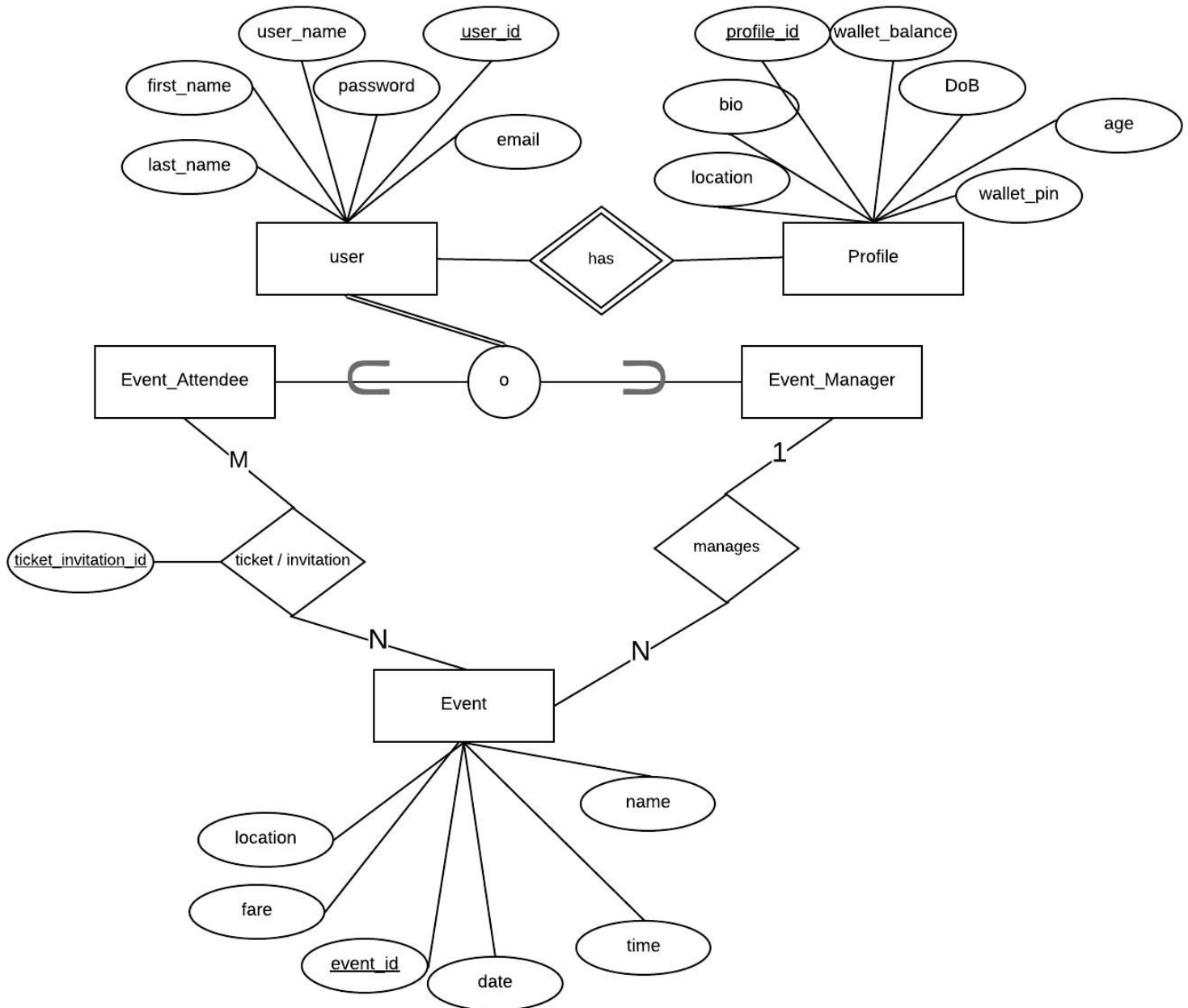
## VI. Withdraw Money



## VII. Add Money

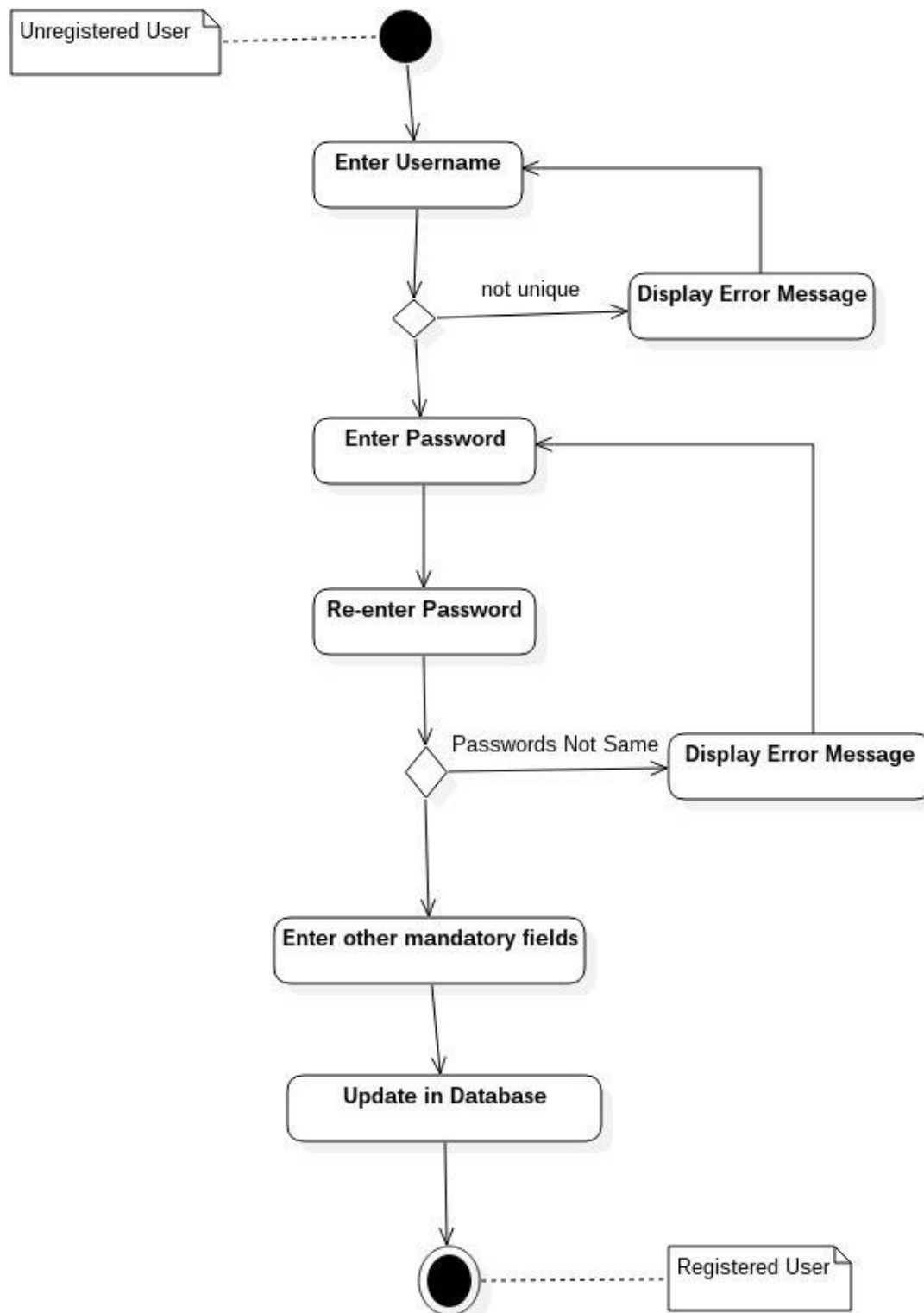


### 3. ER Diagram

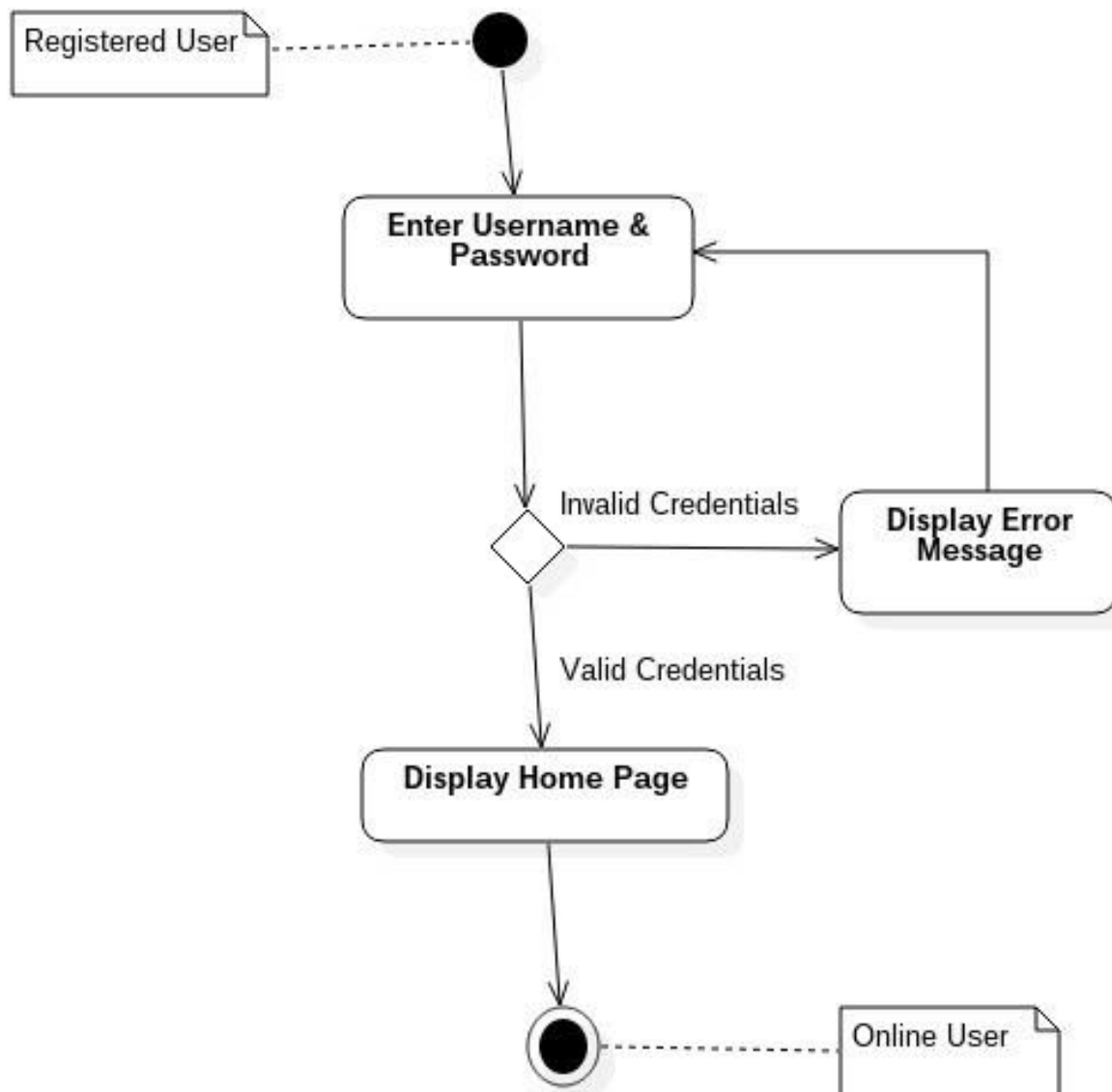


#### 4. Activity Diagram

##### I. Register

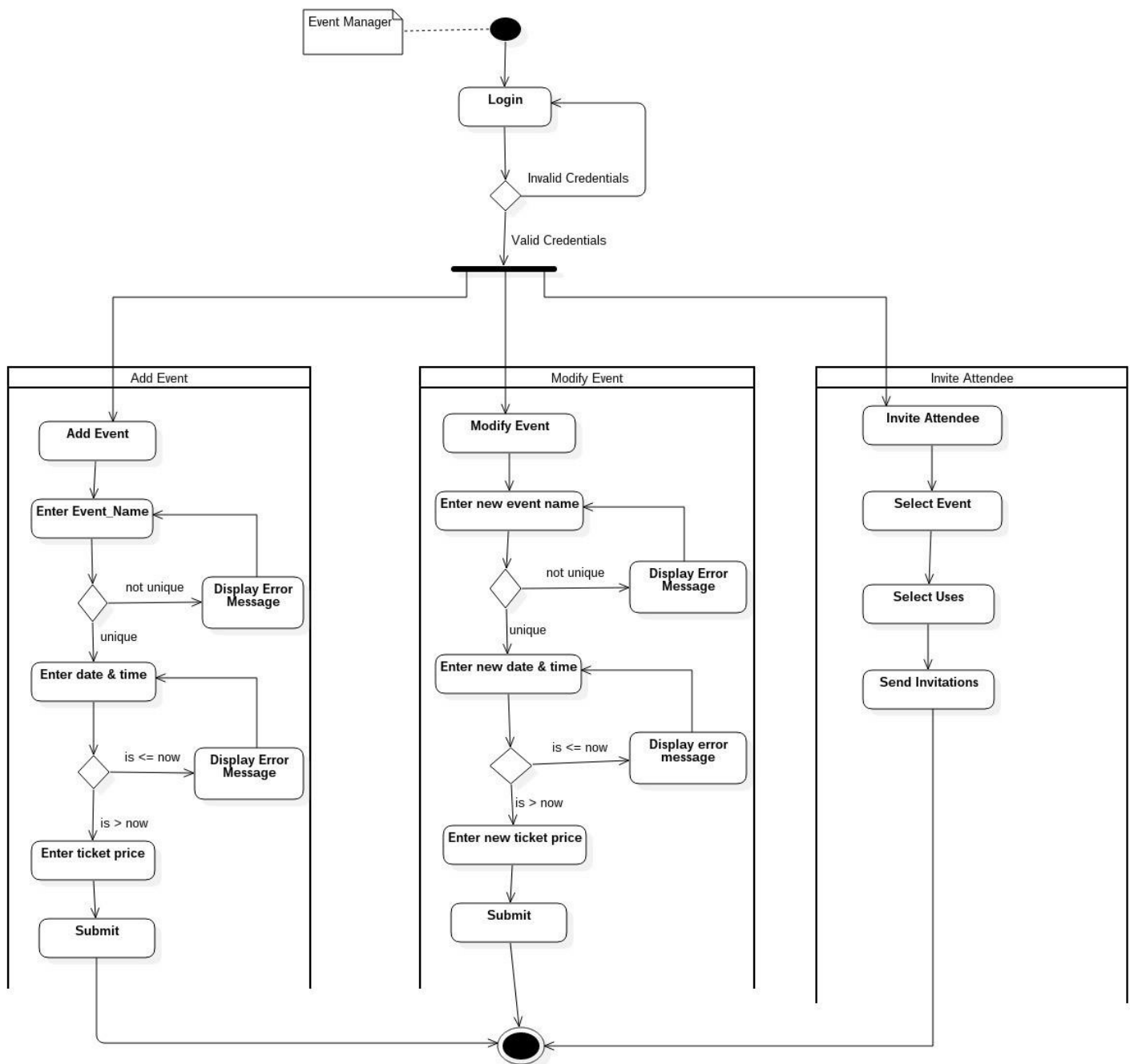


## II. Login

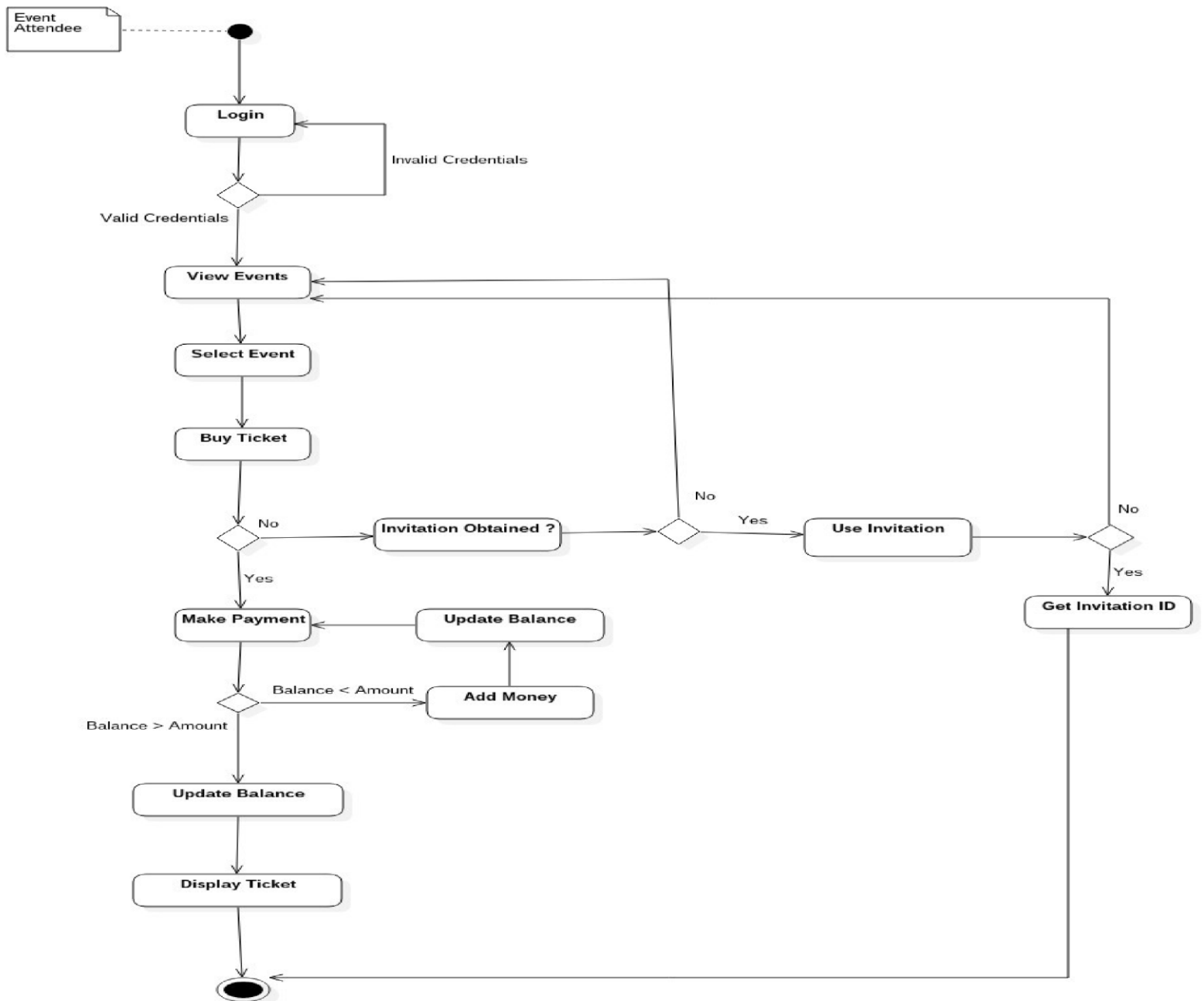




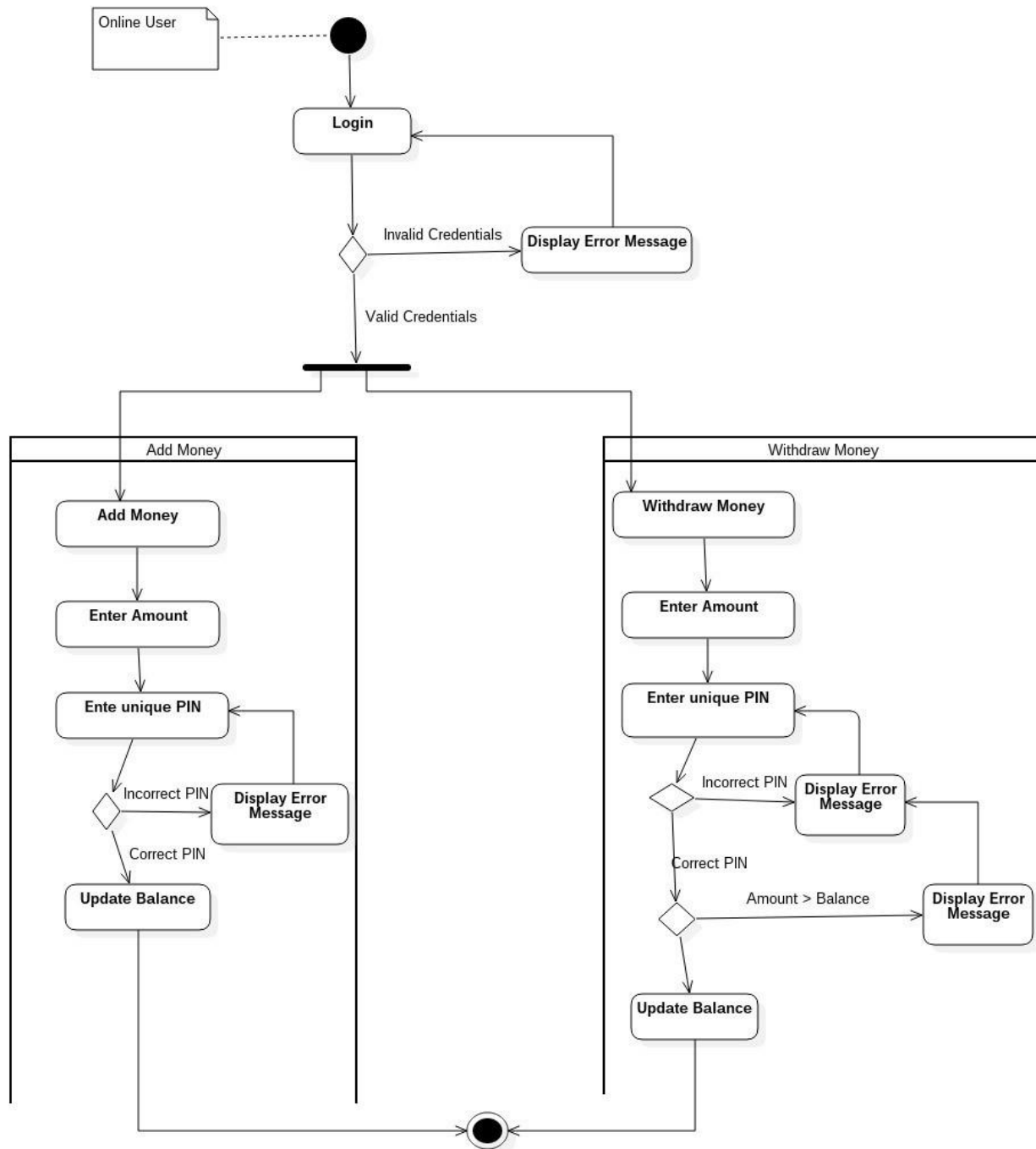
### III. Event Manager



## IV. Event Attendee



## V. Add / Withdraw Money



## **Appendix C: To Be Determined List**

- 2.6 – User Documentation
- 3.1 – User Interfaces
- 3.2 – Hardware Interfaces
- 3.4 – Communication Interfaces
- 6 – Other Requirements