

**OBJECT ORIENTED SOFTWARE**  
**ENGINEERING**  
**SE202a**

**LAB MANUAL**



Submitted by: **Baladitya Dhaka(2K21/SE/51)**  
& **Ashish Bhushan(2K21/SE/41)**

Submitted to:

**Department of Software Engineering**

**Delhi Technological University**  
**Shahbad Daulatpur, Main Bawana Road, Delhi-**  
**110042**

## **INDEX**

S.NO.	EXPERIMENT	DATE	PAGE NO.	SIGNATURE
1	Problem statement			
2	IRD			
3	Use Case diagram			
4	Use Case description			
5	SRS			
6	Class Diagram			
7	Sequence Diagram			
8	Activity Diagram			
9	State-chart Diagram			
10	Test case matrixes			

## **EXPERIMENT-1**

**AIM-** Write a problem statement for event website.

### **PROBLEM STATEMENT**

In this day and age, everyone wants to be updated with the latest of the information of event. The event website is used to manage all the activities related to events such as singing events, comedy events, movie shows, etc. The main aim is to provide the user with latest events and a location filter. The success of the event depends heavily on the effectiveness of its website, and it is critical that this problem is addressed to ensure a successful and profitable event. Current website have many limitations such as-

1. Current websites usually have problems in recognizing nearby locations.
2. Current website doesn't allow to show previous reviews of the events for a feedback for the attendees.
3. The current event website is not effectively engaging and informing potential attendees, resulting in low ticket sales and a lack of excitement around the event.
4. The website design and content do not effectively communicate the unique value proposition of the event, and key information such as schedule, location, and pricing is difficult to find and understand
5. No or problems in refund and payment gateway issues.
6. Number of attendees are not fixed and creates overcrowding.
7. High network traffic during registration for big events causes crashes.

The main features of the website will be:

- Website will provide detailed information about the event like the date, time, location , and other details of the events.
- Website will have an efficient and reliable location access to sort nearby events.
- Website will provide a brief overview of the events
- Quick refund if applicable in case.
- Website will include a helpdesk/contact information for queries and help.
- It will allow for users to write reviews about the previous events for feedback about the event.

## **EXPERIMENT-2**

**AIM-** make a IRD(initial requirement document) for event website

### **IRD (Initial Requirement Document)**

Title of the project	Event website
Stakeholders involved in capturing requirements	Organiser and Attendees
Techniques used for requirement capturing	Brainstroming
Name of the persons along with designation	Baladitya and Ashish
Date	25/1/2023
Version	1.0

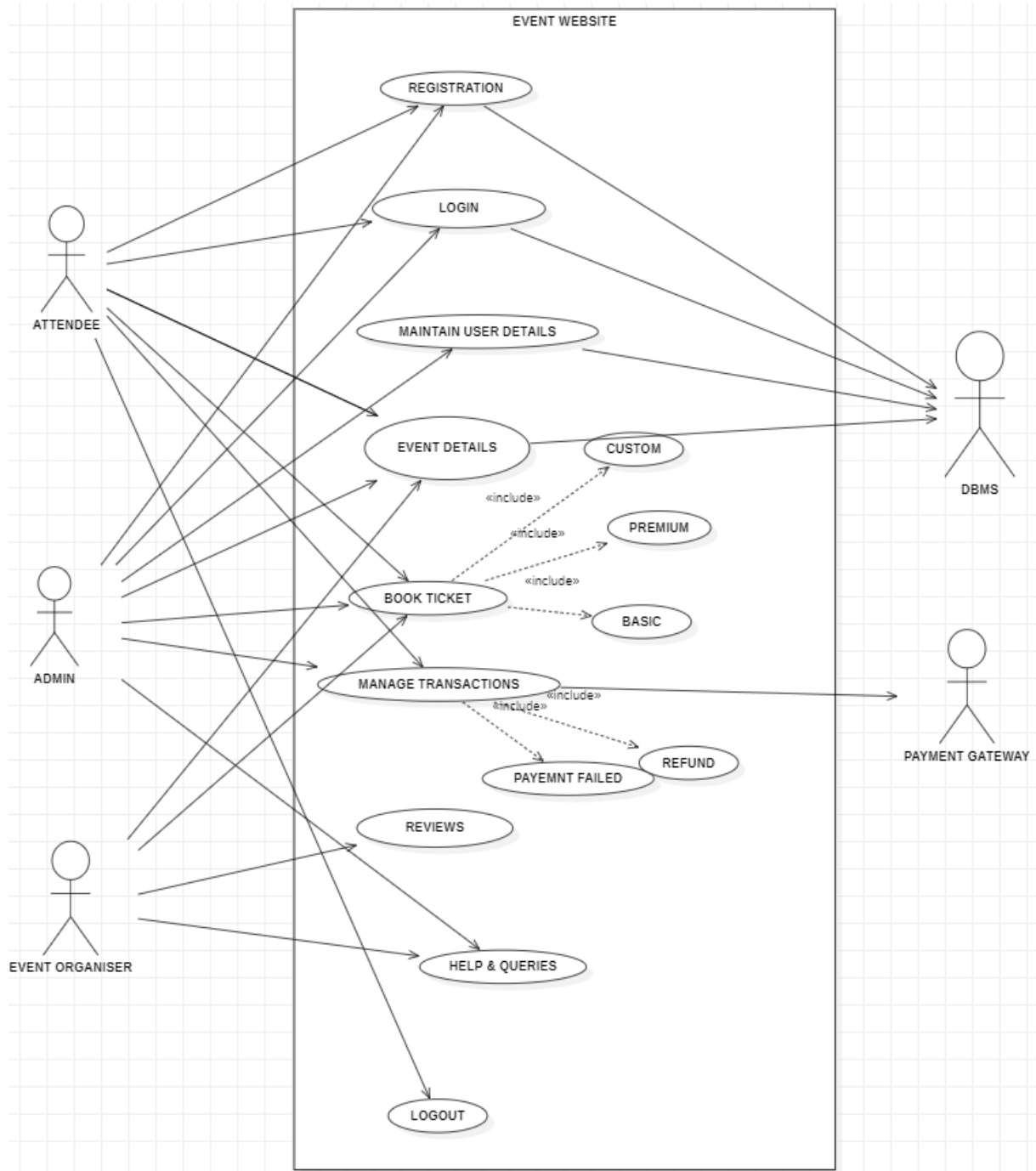
#### **List of initial requirement document**

1. A website is to be implemented that can run on internet and most of the mainstream browsers.
2. The system shall be able to generate LOGIN ID and password .
3. There should be 2 types of member – attendee and organiser.
4. The admin shall be able to maintain the details of all the events.
5. The admin shall be able to maintain details of all the attendees of the events.
6. If the event requires tickets, then it shall be able to provide information about the tickets and their price.
7. Website should be able to accept tickets through a registration and payment gateway.
8. Website should be able to provide the availability of tickets.
9. Website should be able to provide detailed information about the event like the date , time, location, and other details of that event.
10. Information of all the events should be easy to find and easy to read and include a emergency contact information for attendees in case of emergencies.
11. If the events has multiple sessions , it should be able to provide a schedule of event on the website.
12. Website should provide the rules and regulations of the event that attendees need to be aware of , such as dress code, age restriction, etc.it should include these information on the website.
13. Website should include a easy to find contact/query/help page where phone number, email, other relevant contact information.
14. Website should have a payment gateway to accept payments and provide refunds.
15. Attendee should be able select there following location and browse through nearby events .
16. Attendees can search and sort for the events of there liking such as music, comedy shows ,etc.
17. Attendees can contact the helpdesk for any help, queries and support.
18. Event organiser should have a list of all events and attendees.
19. Event organiser should have system that lhas the seating arrangements based on there pricing.
20. Event organiser should be able to reply to any queries of the attendees.

### EXPERIMENT-3

**AIM-** make a Use case diagram for event website using Staruml .

#### **USE CASE DIAGRAM**



## **EXPERIMENT-4**

**AIM-** Make a usecase description for event website

### **Use case description for registration**

	<b>Brief description:</b> This use case allows an actor to register securely in the system Creating and saving data in the database
	<b>Actor:</b> attendee and admin
	<b>Precondition:</b> none
	<b>Postcondition:</b> the attendee is successfully registered and his details are securely saved
	<b>Basic flow:</b> <ol style="list-style-type: none"><li>1. The use case starts when the user clicks the “Sign Up” tab on the navigation bar.</li><li>2. The system displays the signup page that allow users to fill in their <b>username/email address, first and last name</b> and their <b>password</b>.</li><li>3. The user must enter a valid username that does not already exists.</li><li>4. The user keys in the details and clicks “Sign Up” button.</li><li>5. The system displays the user’s profile page.</li></ol>
	<b>Alternate flow:</b> <ol style="list-style-type: none"><li>1. <b>Invalid details-</b> if the user id/name entered by the user is already taken/exists or the information entered by the user is missing then the registration fails and the user is redirected to basic flow to register again.</li><li>2. <b>User already registered-</b> if the user trying to register is already registered then the login is failed and the user is redirected to login page where he/she must enter details to login in.</li></ol>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

### **use case description for login**

	<b>Brief description:</b> user tries to sign into the system to view there details.
	<b>Actor:</b> attendee and admin
	<b>Precondition:</b> user should be registered before hand in the database
	<b>Postcondition:</b> the user has access to the system
	<b>Basic flow:</b> <ol style="list-style-type: none"><li>1. The use case starts when the user clicks the “Login” tab on the navigation bar</li><li>2. The system displays the login page that allow users to fill in their username/email address and password</li><li>3. The user keys in the details and clicks “Sign In” button</li><li>4. The system checks the details in the database</li><li>5. The system displays the user’s profile page</li></ol>
	<b>Alternate flow:</b> <ol style="list-style-type: none"><li>1. <b>Invalid credentials-</b> if the user name/id or password entered by the user is incorrect or missing then the login fails and the user is redirected back to the basic flow.</li><li>2. <b>User not registered-</b>if the user trying to login , has no saved data in the database then the user is not registered and is redirected to the registration</li></ol>

	page.
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

### use case description for maintain user details

	<b>Brief description:</b> this use case documents the steps that the admin must follow in order to maintain user details and add, update ,delete, and view user information
	<b>Actor:</b> admin
	<b>Precondition:</b> admin must be logged into the system before this use case begins
	<b>Postcondition:</b> if the use case is successful, then user information is added, updated , deleted or viewed.
	<p><b>Basic flow:</b></p> <ol style="list-style-type: none"> <li>1. The use case starts when the admin wished to add/update/delete/view user information.</li> <li>2. Once the admin provides the requested information, one of the following subflows is executed- <ul style="list-style-type: none"> <li>• if the admin selects “ add user ”, the add a user subflow is executed</li> <li>• if the admin selects “update user details ”,the update user details subflow is executed</li> <li>• if the admin selects “delete user details ”,the delete user details subflow is executed</li> <li>• if the admin selects “view user details ”,the view user details subflow is executed</li> </ul> </li> </ol> <p><b>basic flow 1(add a user)</b> the system requests that the admin enter the user details. this includes-</p> <ol style="list-style-type: none"> <li>1. user number</li> <li>2. photograph</li> <li>3. name</li> <li>4. Date of Birth</li> <li>5. Email</li> <li>6. Address</li> <li>7. Login id</li> <li>8. Password</li> </ol> <p>Once the administrator provides the requested information, the system checks that user number is unique. Then user is added to the system.</p> <p><b>Basic flow 2(Update a User)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator enter the user number.</li> <li>2. The administrator enters the user number.</li> <li>3. The system retrieves and displays the user’s information.</li> <li>4. The administrator makes the desired changes to the user information. This includes any of the information specified in the Add a user subflow.</li> <li>5. Once the administrator updates the necessary information, the system updates the user record with the updated information.</li> </ol> <p><b>Basic Flow 3(Delete a User)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the user number of the user.</li> <li>2. The administrator enters the user number. The system retrieves and displays the</li> </ol>

	<p>user information.</p> <ol style="list-style-type: none"> <li>3. The system prompts the Administrator to confirm the deletion of user record.</li> <li>4. The administrator verifies the deletion.</li> <li>5. The system deletes the record.</li> </ol> <p><b>Basic Flow 4(View a User)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the user number.</li> <li>2. The system retrieves and displays the user information.</li> </ol>
	<p><b>Alternate flow:</b></p> <ol style="list-style-type: none"> <li>1. <b>invalid entry-</b> if the admin enters invalid accession number number/id the system displays an error message. the admin returns to basic flow and re enter the accession number/id.</li> <li>2. <b>User not found-</b> if in the update a user, delete or view user details, the user information with the specified accession number/id does not exist ,the system displays a error message. the admin returns to basic flow and may re-renter the accession number.</li> <li>3. <b>Deletion not allowed-</b>if in the delete user details of the user selected ,ongoing ticket details or registration details are present, then the admin is not allowed to delete the user information and basic flow is started again</li> </ol>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

### use case description for event details

	<b>Brief description:</b> this use case provides all the details about the events
	<b>Actor:</b> attendee, admin and event organiser
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the events and location details.
	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>1. The use case starts when the user clicks the “events details” tab on the navigation bar</li> <li>2. The system displays the details of the events that allow users to purchase tickets</li> <li>3. The user is shown information of events on the basis of his/her location</li> </ol> <p><b>Basic Flow 1 (Add an Event)</b></p> <p>The system requests that the administrator enter the event information. This includes:</p> <ul style="list-style-type: none"> <li>• Accession number</li> <li>• Event organiser</li> <li>• ISBN</li> <li>• Title</li> </ul>



	<ul style="list-style-type: none"> <li>• Photographs</li> <li>• performances</li> <li>• Artists</li> </ul> <p>Once the administrator provides the requested information, the event is added to the system.</p> <p><b>Basic Flow 2 (Update an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator enter the accession number.</li> <li>2. The administrator enters the accession number.</li> <li>3. The system retrieves and displays the event information.</li> <li>4. The administrator makes the desired changes to the event information. This includes any of the information specified in the Add a event subflow.</li> <li>5. Once the administrator updates the necessary information, the system updates the event information with the updated information.</li> </ol> <p><b>Basic Flow 3 (Delete an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the accession number.</li> <li>2. The administrator enters the accession number. The system retrieves and displays the required information.</li> <li>3. The system prompts the administrator to confirm the deletion of the event record.</li> <li>4. The administrator verifies the deletion.</li> <li>5. The system deletes the record.</li> </ol> <p><b>Basic Flow 4 (View an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the accession number.</li> <li>2. The system retrieves and displays the event information.</li> </ol>
	<p><b>Alternate flow:</b></p> <ol style="list-style-type: none"> <li>1. Login failed-if the user is not logged into the system, the system will show an error and redirects to the login page in order to login again.</li> <li>2. Location access-if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again.</li> </ol>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

### use case description for book tickets

	<b>Brief description:</b> this use case provides all the details about the tickets and purchase tickets for the corresponding events.
	<b>Actor:</b> attendee, admin and event organiser
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the events and location details.
	<p><b>Basic flow:</b></p> <ol style="list-style-type: none"> <li>1. The use case starts when the user clicks the “events details” tab on the navigation bar</li> <li>2. The system displays the details of the events that allow users to purchase tickets</li> <li>3. The user is shown information of events on the basis of his/her location</li> </ol>

	<b>Alternate flow:</b> 3. <b>Login failed</b> -if the user is not logged into the system the the system will show an error and redirects to the login page in order to login again. 4. <b>Location access</b> -if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again.
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b>

### use case description for manage transaction

	<b>Brief description:</b> this use case manages the transaction made by the attendees to purchase event tickets/passes.
	<b>Actor:</b> attendee, admin
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the event payments
	<b>Basic flow:</b> 1. The use case starts when the user clicks the “payment” tab on the navigation bar 2. The system displays the details of the events that allow users to purchase tickets 3. The user is shown information of payment and amount to be paid 4. The user is shown the price and other details of the events. 5. The user then selects payment gateway and payment methods 6. The user then completes the payment.
	<b>Alternate flow:</b> <b>1.Login failed</b> -if the user is not logged into the system then the system will show an error and redirects to the login page in order to login again. <b>2.Location access</b> -if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again. <b>3.Payment failed</b> -if the payment is failed due to some fault, then the user is prompted with a failed payment error with the fault specification and redirected to the payment page.
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> payment failed and refund

## **EXPERIMENT-5**

**AIM-** Make an SRS(Software Requirement Specification) document for event website.

### **Software Requirement Specification**

#### **1. Introduction**

##### **1.1 Purpose**

The purpose of this document is to define the functional and non-functional requirements for an event management website. The website will allow users to browse, search, and book tickets for various events. The system will be designed to provide a seamless user experience, secure payment transactions, and easy event management for administrators.

##### **1.2 Scope**

###### **Dos**

- issue of login id and [password to system operators
- maintain event details and details of event organiser
- User maintains details of attendees in the event to provide tickets
- issues tickets to attendees and their information
- maintain details of seats and reserve seats and bookings
- Ensure user data privacy and security
- Provide excellent customer support to users
- Generates reports, tickets and passes.
- Provides a search facility for nearby events.

###### **Don'ts**

- Do not use overly complicated language or design
- Do not require unnecessary personal information from users
- Do not overload the website with too much content
- Do not have a confusing or difficult-to-use user interface

###### **Benefits**

- Increased accessibility to attendees for events nearby
- Flexibility for attendees to browse at their own pace and on their own schedule
- Increased reach for event organisers, as they can reach a wider audience with their events.
- Convenience for both attendees and event organiser, as they can participate in events from anywhere with an internet connection.

#### **a) Definitions, acronyms and abbreviations**

- **EW:** event website
- **SRS:** software requirement specifications
- **System operator:** System administrator, attendee, event organiser
- **RAM:** Random access memory
- **Login id:** It is a unique sequence ID allocated to each attendee or organiser in the catalog.
- **attendee:** Any candidate admitted in a programme offered by a school.

- **System administrator/Administrator:** User having all the privileges to operate the EMW.
- **Event organiser:** actor that organises the event.

## 1.4 References

a.

## 1.5 Overview

The rest of SRS document describes various system requirements, interfaces, features and functionalities. The system will be designed to provide a seamless user experience, secure payment transactions, and easy event management for administrators. The website should allow users to register and create an account. Users should be able to login and logout securely.

## 2. Overall description

- The EW maintains records of events in nearby location of the attendee, info of event organiser and ticket info. The system admin will receive lists of the admitted attendees. Then the info of the attendees is sent to the event organiser.
- The EW allows attendees to purchase tickets and book seats to event. It provides a payment gateway through which it will collect payments from the attendees. It generates reports , tickets and passes for the gattendees and gives info about this to then event organiser. It allows the attendees to give feedback of the event they attended and post reviews for the future attendees. It also includes a help and query desk for any customer support required.
- This system is being implemented to facilitate communication between the clients and the potential event-goers. As well, as provide up to the minute updates and information about the event.
- The admin will maintain the following-

- 1-event details of all the events from the catalog
- 2-attendees details attending the event and the event organiser
- 3-information about all the tickets purchased

- Attendee can access the following functions-

- 1- Register and Login
- 2- View event details for nearby events
- 3- Book tickets and passes for the events
- 4-Attendee can use the payment to complete payment of their purchase.
- 5- Attendees can post reviews and feedback for the event
- 6- Attendees can access helpdesk for any information and query

- The admin/attendee/event organiser will require following information from the system:

- 1-user details of all the attendees
- 2-event details
- 3-transactions details and generate reports for bill, ticket, etc.

## **2.1 Product perspective**

The Event Website is a standalone web application that is accessible via a web browser. Event website is intended to be a stand-alone product and should not depend on the availability of other software. The website will be accessible to users worldwide, and will be designed to be scalable and reliable. It shall be compatible with all the mainstream browsers such as chrome, firefox , opera, brave.

### **2.1.1 System interfaces**

NONE

### **2.1.2 User interfaces**

The Event website will have the following user-friendly and menu-driven interfaces:

- (a) Login: to allow the entry of attendees and organiser through valid login ID and password.
- (b)Event Details: to maintain event details.
- (c) User Details: to maintain details of every user that includes attendees and organiser.
- (d) Book ticket/pass: to allow the attendees to purchase a ticket or a pass for the following event
- (e)manage transactions: to allow the attendees to pay for the purchase and refund for cases.
- (f)reviews: to allow the attendees to post a review or a feedback to event they attended.
- (g)helpdesk: to allow user to communicate with the admin or organiser in-case they have any query or need help.

The website should generate the following information:

- (a) Details of all the users using the website
- (b) Details of all the events
- (c) Details of the transactions of tickets/passes.

### **2.1.3 Hardware interfaces**

As the Event Management System is a web-based platform, there are no specific hardware interfaces required to run the system. However, users will need to have access to a computer or mobile device with a modern web browser and reliable internet connectivity to use the system. The system shall be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge. Additionally, the system shall be designed using a responsive design approach to ensure that it is accessible on a range of screen sizes, including desktops, laptops, tablets, and smartphones.

### **2.1.4 Software interfaces**

NONE

### **2.1.5 Communications interfaces**

Communication is via a online network.

### **2.1.6 Memory constraints**

NONE

### **2.1.7 Operations**

The Management of Event Operations: project management, planning and customer satisfaction provides an introduction to the management

of operations for the event planner and venue provider.

#### **2.1.8 Site adaptation requirements**

The system shall support multiple languages to cater to a global audience. The system shall be designed to support language translations of user interface elements and content.

### **2.2 Product functions**

The EW will allow access to registered users with specific roles. Depending upon the user's role, he/she will be able to access only specific modules of the system.

A summary of major functions that the EW will perform is given as follows:

- Protect the database by requiring a correct and registered username and password.
- Make data organization easier by classifying participants according to sub-types of personal events.
- Facilitate a systematic process of entering, organizing, retrieving, modifying and deleting data from the database without the need to go the database itself.
- Add new user information easily.
- Provide an option for users to update information.
- Delete existing client information.
- Create and display new events.
- Provide an easy function where you can go back one form whenever necessary.
- Add new organiser contacts with which future collaboration is expected.
- Display user information in an organized manner for easy understandability.
- Display payment terms of attendee including the total ticket/pass fee, amounts paid by the attendee and the balance. Also provide refund in particular case.

### **2.3 User characteristics**

- Qualification: At least comfortable with one language.
- Experience: Should be well versed/informed about the registration process of the website.
- Technical Experience: Elementary knowledge of computers and browsers.

### **2.4 Constraints**

- Requires a stable internet connection for accessing the website
- As a website is dependent on browser and browser code are different for different browsers. In website development, cross-browser compatibility is important. Website will only operate fully on mainstream browsers. Some functions may be restricted in certain browsers

### **2.5 Assumptions and dependencies**

- Website requires the location access of the user to display nearby events for the attendee.
- Full working EW depends on the browser and internet connection.

- All the event details in downloaded or uploaded by the event organiser and shown to the attendee.

## 2.6 Apportioning of requirements

Not required

## 3. Specific requirements

This section contains the software requirements in detail along with the various forms to be developed.

### 3.1 External interfaces

#### 3.1.1 User interfaces

The following user interfaces will provided by the system:

##### (i) Login Form

This will be the first form, which will be displayed. It will allow the user to access the different forms based on his/her role. Various fields available on this form will be:

- **Login ID:** Alphanumeric of length in the range of 4 to 15 characters. Special characters and blank spaces are not allowed.
- **Password:** Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed.

##### (ii) Change Password

The change password form facilitates the user to change the password. Various fields available on this form will be:

- **Login ID:** Alphanumeric of length in the range of 4 to 15 characters. Special characters and blank spaces are not allowed.
- **Old Password:** Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed.
- **New Password:** Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed.
- **Confirm Password:** Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed. The content of this field must match with the contents of the new password field.

##### (iii) Maintain User Details

This form will be accessible only to the system administrator. It will allow him/her to add/update/delete/view information about users for the website.

Various fields available on this form will be:

- **User number:** Numeric and will have value from 10 to 99999.
- **Photograph:** Will allow user to upload their image.
- **Name:** Alphanumeric, with length 3 to 50 characters. Blank spaces are allowed. Special characters are not allowed.
- **Date of birth:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.
- **Address:** Alphanumeric of length up to 10 to 200 characters. Blank spaces are allowed.
- **Telephone:** Numeric and can have length up to 11 digits.
- **Email:** Alphanumeric and can have length up to 50 characters. Email must have one '@' and '.' symbol.
- **Login date:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.

- **Password:** Alphanumeric of length in the range of 4 to 15 characters. Blank spaces are not allowed. However, special characters are allowed. Initially contains 8 digits randomly generated number (auto-generated).

#### (iv) **Maintain Event Details**

This form will be accessible only to the system administrator. It will allow him/her to add/update/delete/view information about events and its details given from the organiser.

Various fields available on this form will be:

- **Accession number:** Numeric and will have value from 10 to 99999.
- **Event Organiser:** Will display all the organisers.
- **ISBN:** Numeric
- **Title:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
- **Photographs:** will allow the organiser to upload images for the basic overview of the event for the attendee.
- **Performances:** list of all the subevents in the events and the artists performing

#### (v) **Booking Ticket/Passes**

This form will be accessible only to the system administrator and attendee. It will allow him/her to issue tickets/passes to the attendee(s).

Various fields available on this form will be:

- **User number:** Numeric and will have value from 100 to 5999.
- **Accession number:** Numeric and will have value from 10 to 99999.
- **Event:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
- **Ticket /pass type** (the following user information will be displayed):
  - **Basic:** Alphanumeric of length 3 to 100 character.
  - **Premium:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Custom:** customizable as per the attendee needs. It will be of format Alphanumeric of length 3 to 100 character.
- **Booking status** (the following fields will be displayed)
  - **Accession number:** Numeric and will have value from 10 to 99999.
  - **Title:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **booking date:** Will be of format mm/dd/yyyy. It will have 10 alphanumeric characters.
  - **Date of the event:** Date of the event. It will be of format mm/dd/yyyy and will have 10 alphanumeric characters

#### (vi) **Transactions and Payment**

This form will be accessible to the system administrator and attendee. It will allow him/her to pay for the purchase of the ticket/passes for the event.

Various fields available on this form will be:



- **User number:** Numeric and will have value from 100 to 5999.
- **Accession number:** Numeric and will have value from 10 to 99999.
- **Event:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
- **Ticket /pass type** (the following user information will be displayed):
  - **Basic:** Alphanumeric of length 3 to 100 character.
  - **Premium:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Custom:** customizable as per the attendee needs. It will be of format Alphanumeric of length 3 to 100 characters.
- **Payment details** (the following fields will be displayed)
  - **Accession number:** Numeric and will have value from 10 to 99999.
  - **Title:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Select gateway:** It will have 10 alphanumeric characters.
  - **Amount payable:** Numeric and will have value from 10 to 99999.
  - **Select method:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Balance:** amount left to be paid after the payment. Numeric and will have value from 10 to 99999.
  - **Payment confirmation:** whether payment is received or not. Alphanumeric of length 3 to 100 characters
  - **Date of the event:** Date of return. It will be of format mm/dd/yyyy and will have 10 alphanumeric characters

**(vii) Refund**

This form will be accessible to the system administrator and attendee. It will allow him/her to get refund for the purchase of the ticket/passes for the event.

Various fields available on this form will be:

- **User number:** Numeric and will have value from 100 to 5999.
- **Accession number:** Numeric and will have value from 10 to 99999.
- **Event:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
- **Ticket /pass type** (the following user information will be displayed):
  - **Basic:** Alphanumeric of length 3 to 100 character.
  - **Premium:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Custom:** customizable as per the attendee needs. It will be of format Alphanumeric of length 3 to 100 characters.
- **Payment details** (the following fields will be displayed)
  - **Accession number:** Numeric and will have value from 10 to 99999.
  - **Title:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Gateway:** It will have 10 alphanumeric characters.
  - **Amount:** Numeric and will have value from 10 to 99999.
  - **Method:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.

- **Payment confirmation:** whether payment is received or not. Alphanumeric of length 3 to 100 characters
- **Date of the event:** Date of return. It will be of format mm/dd/yyyy and will have 10 alphanumeric characters
- **Payment status** (status of the payment)
  - **Status:** It will have 10 alphanumeric characters.
  - **Amount:** Numeric and will have value from 10 to 99999.
  - **Fault:** Reason due to which payment failed. Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.
  - **Payment confirmation:** whether payment is received or not. Alphanumeric of length 3 to 100 characters
  - **Date of the event:** Date of return. It will be of format mm/dd/yyyy and will have 10 alphanumeric characters

### (viii) Search Event

This form will be accessible to the system administrator and attendee. It will allow him/her to search about an existing event.

The field available on this form will be:

- **Type text here:** Alphanumeric of length 3 to 100 characters. Special characters (except brackets) are not allowed. Numeric data will be allowed.

### (ix) Generates Reports

The reports will be accessible to the system administrator and attendee. The system will generate different reports according to the specified criteria.

#### (i) Details of attendee

- Name
- User number \_\_\_\_\_
- User details

#### (ii) Details of tickets/passes issued to attendee

- Name of the event
- Date of event
- Attendee Name
- User number \_\_\_\_\_
- Ticket ID
- Type of ticket/pass (custom, basic, premium)
- Date of Birth
- Telephone
- Organiser

#### (iii) Status of payment and bill receipt

- Name of attendee
- Payment id

- Reference number \_\_\_\_\_
- User number \_\_\_\_\_
- Total amount \_\_\_\_\_
- Event name
- Payment method
- gateway
- Balance

### 3.1.1 User interfaces

As stated in section 2.1.3

### 3.1.3 Software Interfaces

As stated in section 2.1.4.

### 3.1.4 Communication Interfaces

None

## 3.2 Functions

### 3.2.1 Registration

#### A. Use case description

	<b>Brief description:</b> This use case allows an actor to register securely in the system Creating and saving data in the database
	<b>Actor:</b> attendee and admin
	<b>Precondition:</b> none
	<b>Postcondition:</b> the attendee is successfully registered and his details are securely saved
	<b>Basic Flow:</b> <ol style="list-style-type: none"> <li>6. The use case starts when the user clicks the “Sign Up” tab on the navigation bar.</li> <li>7. The system displays the signup page that allow users to fill in their username/email address, first and last name and their password.</li> <li>8. The user must enter a valid username that does not already exists.</li> <li>9. The user keys in the details and clicks “Sign Up” button.</li> <li>10. The system displays the user’s profile page.</li> </ol>
	<b>Alternate Flow:</b> <ol style="list-style-type: none"> <li>3. Invalid details- if the user’s id/name entered by the user is already taken/exists or the information entered by the user is missing then the registration fails and the user is redirected to basic flow to register again.</li> <li>4. User already registered- if the user trying to register is already registered then the login is failed and the user is redirected to login page where he/she must enter details to login in.</li> </ol>

	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

## B. Validity Checks

- a) Every user will have a unique login ID.
- b) Login ID cannot be blank
- c) Login ID can only have 4 to 15 characters
- d) Login ID will not accept special characters and blank spaces
- e) Password cannot be blank
- f) Length of password can only be 4 to 15 digits
- g) User Name should not be blank
- h) User Name cannot be blank.
- i) Length of User Name can be of 3 to 50 characters.
- j) User Name will only accept alphabetic characters and blank spaces.
- k) Email cannot be blank
- l) Email cannot be blank.
- m) Email can have up to 50 characters.
- n) Email should contain @ and . characters.
- o) Phone cannot be blank.
- p) Phone cannot include alphabets, special characters and blank spaces.
- q) Phone can be up to 10 digits
- r) Date of birth cannot be blank
- s) Alphabets, digits, hyphen and underscore characters are allowed in the password field.
- t) Password will not accept blank spaces.

## C. Sequencing Information

None

## D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

### 3.2.2 Login

#### A. Use case description

	<b>Brief description:</b> user tries to sign into the system to view their details.
	<b>Actor:</b> attendee and admin
	<b>Precondition:</b> user should be registered beforehand in the database
	<b>Postcondition:</b> the user has access to the system
	<b>Basic Flow:</b> <ol style="list-style-type: none"> <li>6. The use case starts when the user clicks the “Login” tab on the navigation bar</li> <li>7. The system displays the login page that allow users to fill in their username/email address and password</li> <li>8. The user keys in the details and clicks “Sign In” button</li> <li>9. The system checks the details in the database</li> </ol>

	<b>10.</b> The system displays the user's profile page
	<b>Alternate flow:</b> <ol style="list-style-type: none"> <li>3. Invalid credentials- if the user name/id or password entered by the user is incorrect or missing then the login fails and the user is redirected back to the basic flow.</li> <li>4. User not registered-if the user trying to login, has no saved data in the database then the user is not registered and is redirected to the registration page.</li> </ol>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

## B. Validity Checks

- a) Every user will have a unique login ID.
- b) Login ID cannot be blank.
- c) Login ID can only have 4 to 15 characters.
- d) Login ID will not accept special characters and blank spaces.
- e) Password cannot be blank.
- f) Password will not accept blank spaces.
- g) Length of password can only be 4 to 15 digits.
- h) Alphabets, digits, hyphen and underscore characters are allowed in the password field.

## C. Sequencing Information

None

## D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

### 3.2.3 Maintain User Details

#### A. Use case description

	<b>Brief description:</b> this use case documents the steps that the admin must follow in order to maintain user details and add, update, delete, and view user information
	<b>Actor:</b> admin
	<b>Precondition:</b> admin must be logged into the system before this use case begins
	<b>Postcondition:</b> if the use case is successful, then user information is added, updated, deleted or viewed.
	<b>Basic flow:</b> <ol style="list-style-type: none"> <li>3. The use case starts when the admin wished to add/update/delete/view user information.</li> <li>4. Once the admin provides the requested information, one of the following subflows is executed- <ul style="list-style-type: none"> <li>• if the admin selects “add user”, the add a user subflow is executed</li> <li>• if the admin selects “update user details”, the update user details subflow is executed</li> <li>• if the admin selects “delete user details”, the delete user details subflow is executed</li> <li>• if the admin selects “view user details”, the view user details subflow is</li> </ul> </li> </ol>

	<p>executed</p> <p><b>Basic flow 1(Add a User)</b>  the system requests that the admin enter the user details. this includes-  9. user number  10. photograph  11. name  12. Date of Birth  13. Email  14. Address  15. Login id  16. Password  Once the administrator provides the requested information, the system checks that user number is unique. Then user is added to the system.</p> <p><b>Basic flow 2(Update a User)</b>  1. The system requests that the administrator enter the user number.  2. The administrator enters the user number.  3. The system retrieves and displays the user's information.  4. The administrator makes the desired changes to the user information. This includes any of the information specified in the Add a user subflow.  5. Once the administrator updates the necessary information, the system updates the user record with the updated information.</p> <p><b>Basic Flow 3(Delete a User)</b>  1. The system requests that the administrator specify the user number of the user.  2. The administrator enters the user number. The system retrieves and displays the user information.  3. The system prompts the Administrator to confirm the deletion of user record.  4. The administrator verifies the deletion.  5. The system deletes the record.</p> <p><b>Basic Flow 4(View a User)</b>  1. The system requests that the administrator specify the user number.  2. The system retrieves and displays the user information.</p>
	<p><b>Alternate flow:</b></p> <p><b>4. invalid entry-</b> if the admin enters invalid accession number number/id the system displays an error message. the admin returns to basic flow and re-enter the accession number/id.</p> <p><b>5. User not found-</b> if in the update a user, delete or view user details, the user information with the specified accession number/id does not exist, the system displays a error message. the admin returns to basic flow and may re-renter the accession number.</p> <p><b>6. Deletion not allowed-</b>if in the delete user details of the user selected, ongoing ticket details or registration details are present, then the admin is not allowed to delete the user information and basic flow is started again</p>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

## B. Validity Checks

- a) Only the administrator/DEO will be authorized to access the user Details module
- b) Every user will have a unique membership number
- c) User number can only have value from 100 to 5999 digits.
- d) User number will not accept alphabets, special characters and blank spaces.
- e) Every User will have a unique membership number
- f) Date of birth cannot be blank.
- g) Address cannot be blank.
- h) Address can have length up to 10 to 200 characters.
- i) Phone cannot be blank.
- j) Phone cannot include alphabets, special characters and blank spaces.
- k) Phone can be up to 11 digits.
- l) Email cannot be blank.
- m) Email can have up to 50 characters.
- n) Email should contain @ and. characters.
- o) Email cannot include blank spaces.

### C. Sequencing Information

None

### D. Error Handling/Response to Abnormal Situations

If any of the validation/sequencing flows does not hold true, appropriate error message will be prompted to the administrator for doing the needful.

#### 3.2.4 Event Details

##### A. Use case description

	<b>Brief description:</b> this use case provides all the details about the events
	<b>Actor:</b> attendee, admin and event organiser
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the events and location details.
	<p><b>Basic Flow:</b></p> <ol style="list-style-type: none"> <li>4. The use case starts when the user clicks the “events details” tab on the navigation bar</li> <li>5. The system displays the details of the events that allow users to purchase tickets</li> <li>6. The user is shown information of events on the basis of his/her location</li> </ol> <p><b>Basic Flow 1 (Add an Event)</b> The system requests that the administrator enter the event information. This includes:</p> <ul style="list-style-type: none"> <li>• Accession number</li> <li>• Event organiser</li> <li>• ISBN</li> <li>• Title</li> <li>• Photographs</li> <li>• performances</li> </ul>

	<ul style="list-style-type: none"> <li>• Artists</li> </ul> <p>Once the administrator provides the requested information, the event is added to the system.</p> <p><b>Basic Flow 2 (Update an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator enter the accession number.</li> <li>2. The administrator enters the accession number.</li> <li>3. The system retrieves and displays the event information.</li> <li>4. The administrator makes the desired changes to the event information. This includes any of the information specified in the Add a event subflow.</li> <li>5. Once the administrator updates the necessary information, the system updates the event information with the updated information.</li> </ol> <p><b>Basic Flow 3 (Delete an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the accession number.</li> <li>2. The administrator enters the accession number. The system retrieves and displays the required information.</li> <li>3. The system prompts the administrator to confirm the deletion of the event record.</li> <li>4. The administrator verifies the deletion.</li> <li>5. The system deletes the record.</li> </ol> <p><b>Basic Flow 4 (View an Event)</b></p> <ol style="list-style-type: none"> <li>1. The system requests that the administrator specify the accession number.</li> <li>2. The system retrieves and displays the event information.</li> </ol>
	<p><b>Alternate flow:</b></p> <ol style="list-style-type: none"> <li>5. Login failed-if the user is not logged into the system, the system will show an error and redirects to the login page in order to login again.</li> <li>6. Location access-if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again.</li> </ol>
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> N/A

## B. Validity Checks

- a) Only the administrator/DEO will be authorized to access the user Details module
- b) Every event will have a unique accession number.
- c) Accession number cannot be blank.
- d) Accession number can only have value from 10 to 99999 digits.
- e) Event organiser cannot be blank.
- f) ISBN number cannot be blank.
- g) Length of ISBN number for any user can only be equal to 11 digits.
- h) ISBN number will not accept alphabets, special characters and blank spaces.
- i) Title cannot be blank.
- j) Length of title can be of 3 to 100 characters.
- k) Title will only accept alphabetic characters, brackets, numeric digits and blank spaces.
- l) artists cannot be blank.
- m) Length of first name and last name can be of 3 to 300 characters.
- n) Artists will not accept special characters and numeric digits.



### C. Sequencing Information

None

### D. Error Handling/Response to Abnormal Situations

If any of the validation flows does not hold true, appropriate error message will be prompted to the user for doing the needful.

#### 3.2.5 Book Tickets

##### A. Use case description

	<b>Brief description:</b> this use case provides all the details about the tickets and purchase tickets for the corresponding events.
	<b>Actor:</b> attendee, admin and event organiser
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the events and location details.
	<b>Basic Flow:</b> <b>4.</b> The use case starts when the user clicks the “events details” tab on the navigation bar <b>5.</b> The system displays the details of the events that allow users to purchase tickets <b>6.</b> The user is shown information of events on the basis of his/her location <b>7.</b> The user is shown the price and other details of the events.
	<b>Alternate Flow:</b> <b>1. Login failed-</b> if the user is not logged into the system the the system will show an error and redirects to the login page in order to login again. <b>2. Location access-</b> if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again.
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> custom, basic, premium

##### B. Validity Checks

- The administrator and attendee will be authorized to access the book tickets module
- Every user will have a unique membership number.
- user number can only have value from 100 to 5999 digits.
- User number will not accept alphabets, special characters and blank spaces.
- user list cannot be blank.
- Accession number cannot be blank.
- Accession number can only have value from 10 to 99999 digits.
- Date cannot be blank.
- Title cannot be blank.
- Length of title can be of 3 to 100 characters.
- Title will only accept alphabetic characters, brackets, numeric digits and blank spaces.
- Amount cannot be blank.
- Amount can only have numeric value from 10 to 9999999 digits

### C. Sequencing Information

User and event details should be available in the system.

### D. Error Handling/Response to Abnormal Situations

If any of the validation/sequencing flows does not hold true, appropriate error message will be prompted to the administrator for doing the needful.

#### 3.2.6 Manage transaction

### A. Use case description

	<b>Brief description:</b> his use case manages the transaction made by the attendees to purchase event tickets/passes.
	<b>Actor:</b> attendee, admin
	<b>Precondition:</b> user should be logged into the system and must provide his/her location
	<b>Postcondition:</b> the user has access to all information about the event payments
	<b>Basic flow:</b> 1. The use case starts when the user clicks the “payment” tab on the navigation bar 2. The system displays the details of the events that allow users to purchase tickets 3. The user is shown information of payment and amount to be paid 4. The user is shown the price and other details of the events. 5. The user then selects payment gateway and payment methods 6. The user then completes the payment.
	<b>Alternate flow:</b> <b>1.Login failed</b> -if the user is not logged into the system then the system will show and error and redirects to the login page in order to login again. <b>2.Location access</b> -if the user fails to give his/her location based on which the events will be shown the system will show an error message and user will be prompted to give location access again. <b>3.Payment failed</b> -if the payment is failed due to some fault, then the user is prompted with a failed payment error with the fault specification and redirected the payment page.
	<b>Special requirements:</b> N/A
	<b>Associated use case:</b> payment failed and refund

### B. Validity Checks

- a) The administrator and attendee will be authorized to access the book tickets.
- b) Membership number can only have value from 10 to 5999 digits
- c) Transaction id should not be blank
- d) Transaction id can only have value from 10 to 9999.
- e) Transaction id must be unique
- f) Event Title must not be blank
- g) Payment status must not be blank
- h) Payment status can only have alphabetic characters.

### C. Sequencing Information

User, event and payment details should be available in the system.

### D. Error Handling/Response to Abnormal Situations

If any of the validation/sequencing flows does not hold true, appropriate error message will be prompted to the administrator for doing the needful.

### 3.3 Performance Requirements

As the Event Management System is a web-based platform, there are no specific hardware interfaces required to run the system. Responses of the website should be quick and smooth, it should not feel laggy or poorly optimized.

Users will need to have access to a computer or mobile device with a modern web browser and reliable internet connectivity to use the system. The system shall be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge. The system shall be designed using a responsive design approach to ensure that it is accessible on a range of screen sizes, including desktops, laptops, tablets, and smartphones.

### 3.4 Logical Database Requirements

Table name	Description
User profile	Stores information about users
Event information	Stores information about events
Payment information	Stores information regarding payments
Content management	Stores the website content
Refund	Stores information about the refunds
Event organiser	Stores information about the organisers
Attendees	Stores information about the attendees
Reviews	Stores all the reviews about events

### 3.5 Design Constraints

NONE

### 3.6 Software System Attributes

#### 3.6.1 Reliability

This requirement relates to the website's availability and functionality. The website should be available at all times, with minimal downtime and interruptions, to ensure that users can access the information they need and purchase tickets without any issues.

#### 3.6.2 Usability

The website shall provide a user-friendly interface that is easy to navigate and understand. The website shall be designed to be accessible to users with disabilities, such as screen readers and keyboard-only users.

#### 3.6.3 Security

The website shall implement secure measures to protect user data, such as encryption of sensitive data, and protection against common web vulnerabilities, such as cross-site scripting (XSS) and SQL injection attacks.

#### 3.6.4 Maintainability

The application should be designed in a maintainable manner such that it will be easy to incorporate new changes and requirements in the individual modules.

#### 3.6.5 Backup and Recovery

The website should be backed up regularly, and a recovery plan should be in place to restore the website in case of a catastrophic event.

#### 3.6.6 Performance

The website shall be designed to handle high traffic and provide fast response times. The website shall be able to handle concurrent user requests and provide an uptime of at least 99%.

#### 3.6.7 Mobile Optimization

This requirement relates to how well the website performs on mobile devices, such as smartphones and tablets. The website should be designed with a responsive design and mobile-specific features, such as mobile ticket scanning, to ensure that users can access the website and purchase tickets from their mobile devices.

#### 3.6.8 Compatibility

This requirement relates to ensuring that the website is compatible with different web browsers and operating systems. The website should be designed to work on a wide range of devices and platforms, to ensure that it can be accessed by the widest possible audience.

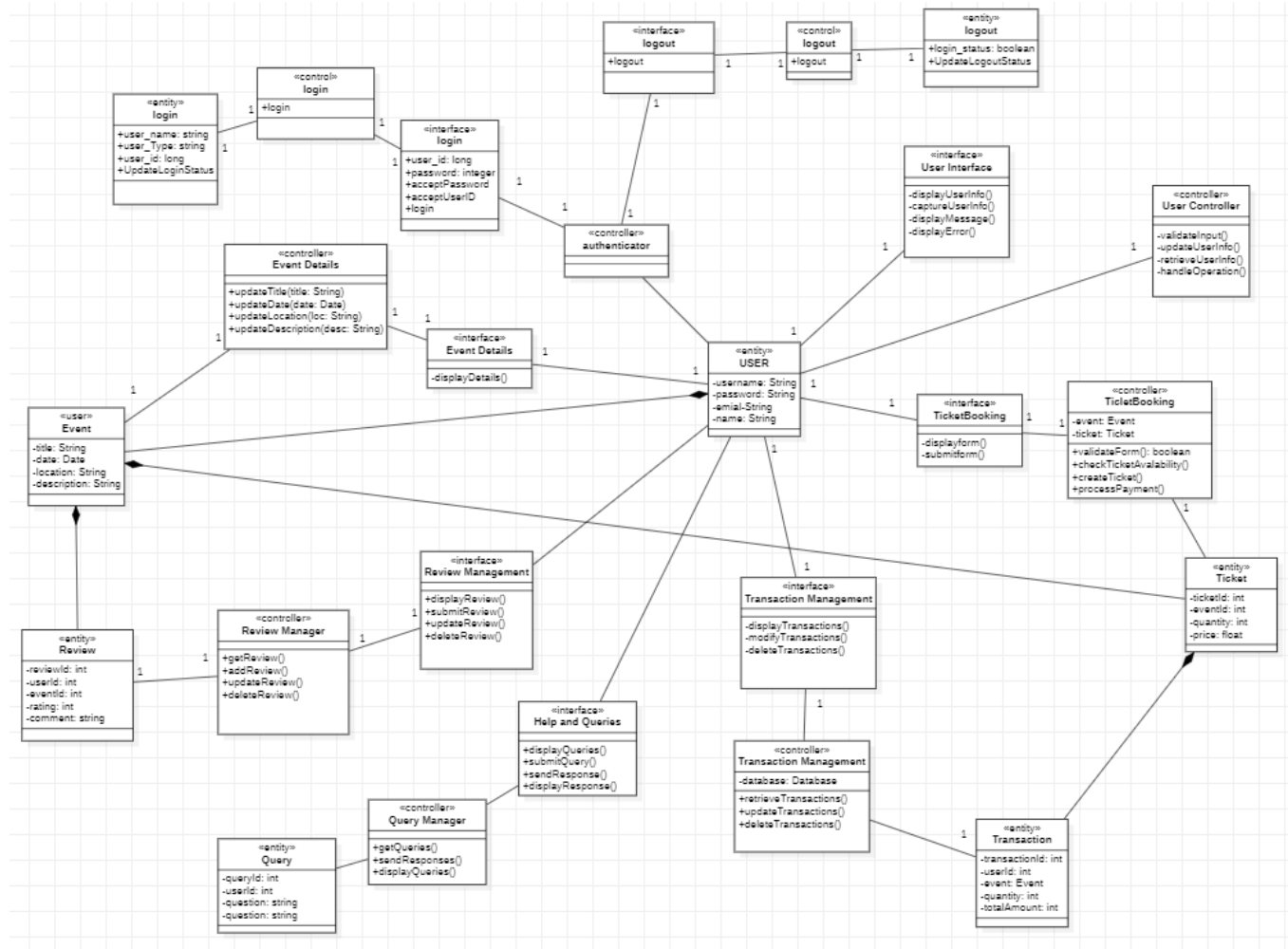
### 3.7 Other requirement

NONE

## EXPERIMENT-6

**AIM-** Make a class diagram for event website using star uml

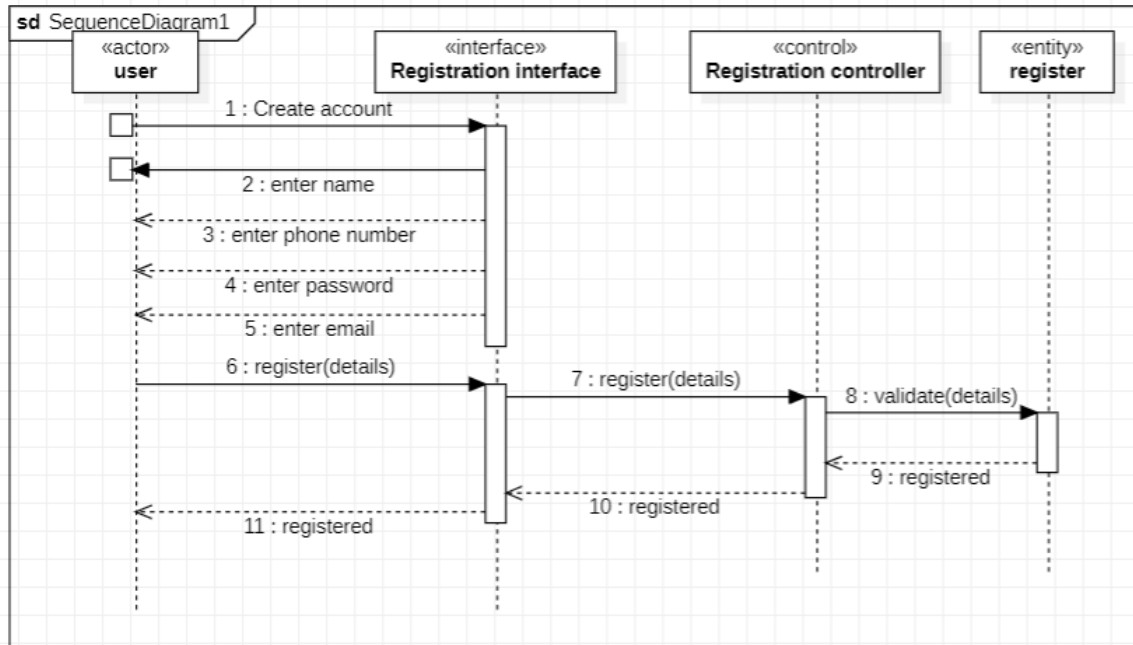
### CLASS DIAGRAM



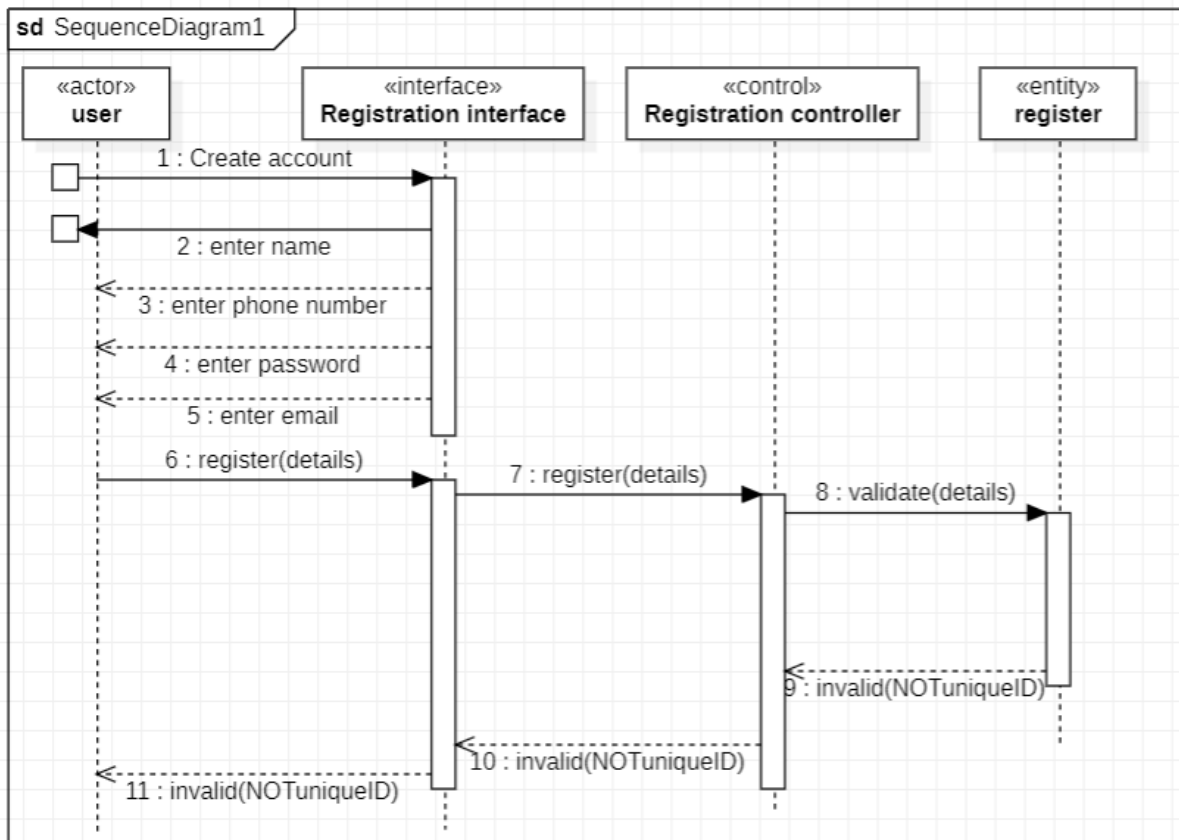
## EXPERIMENT-7

**AIM-** Make sequence diagram for event website using star uml

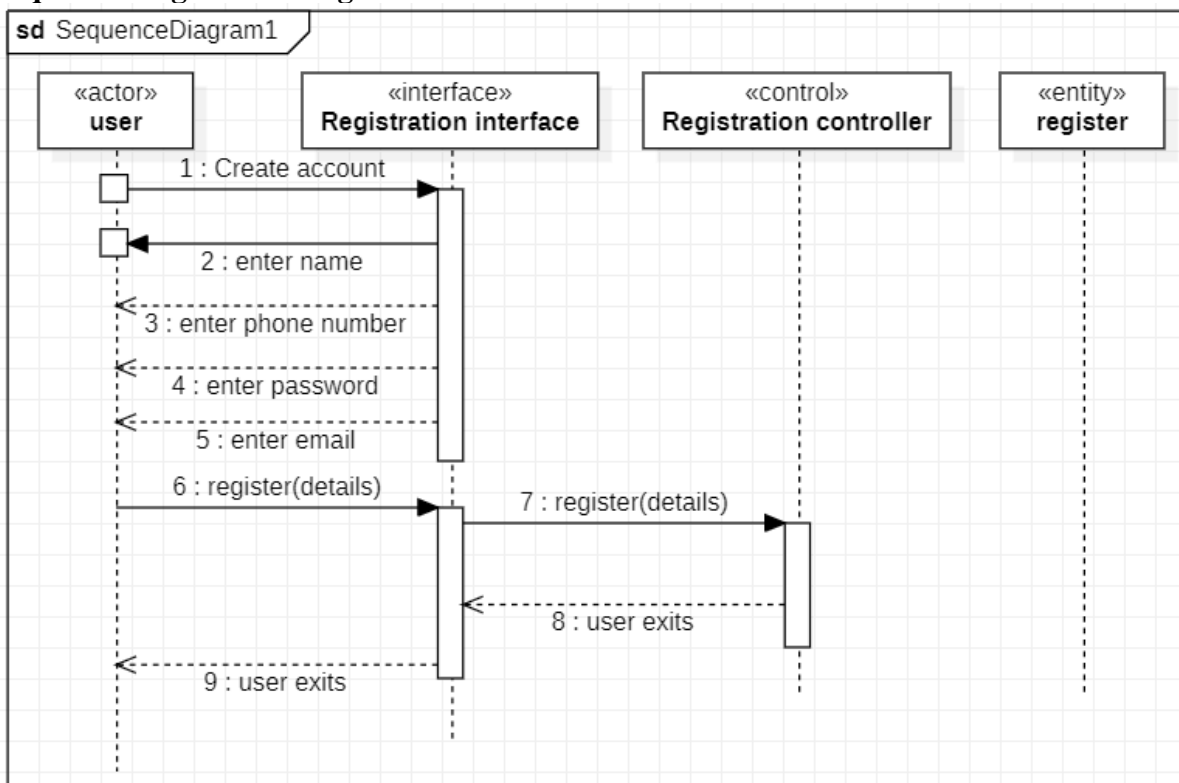
### Sequence diagram for Registration basic flow



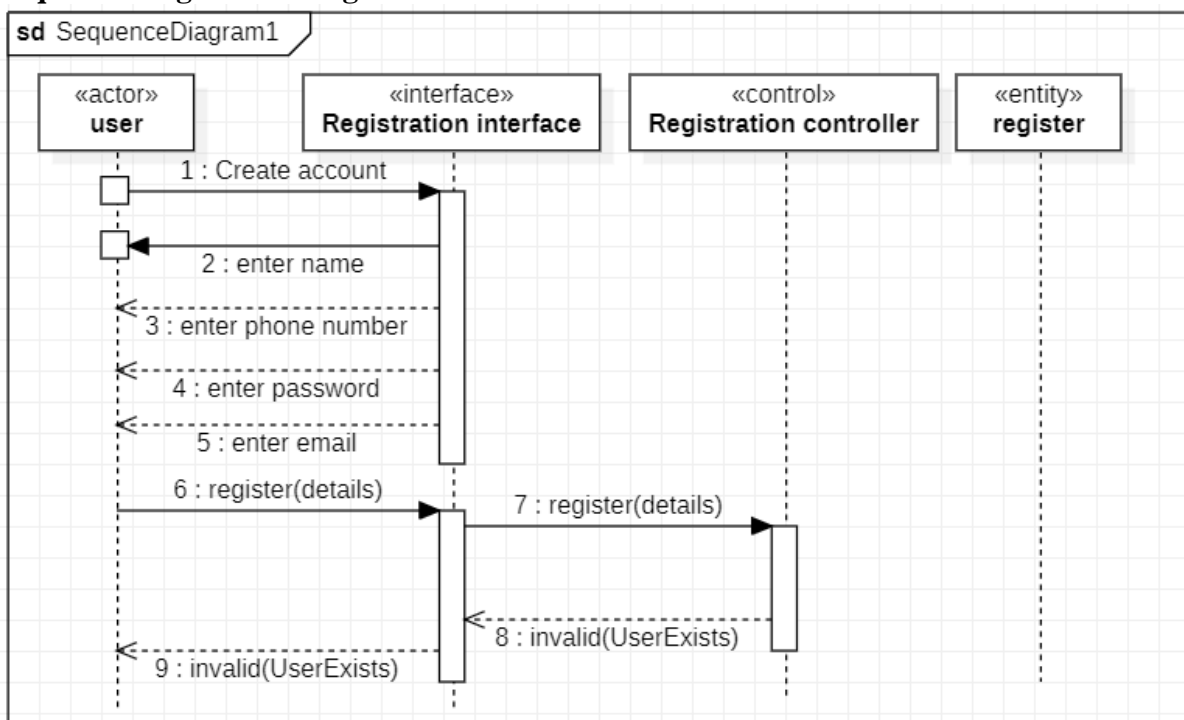
### Sequence diagram for Registration alternate flow 1



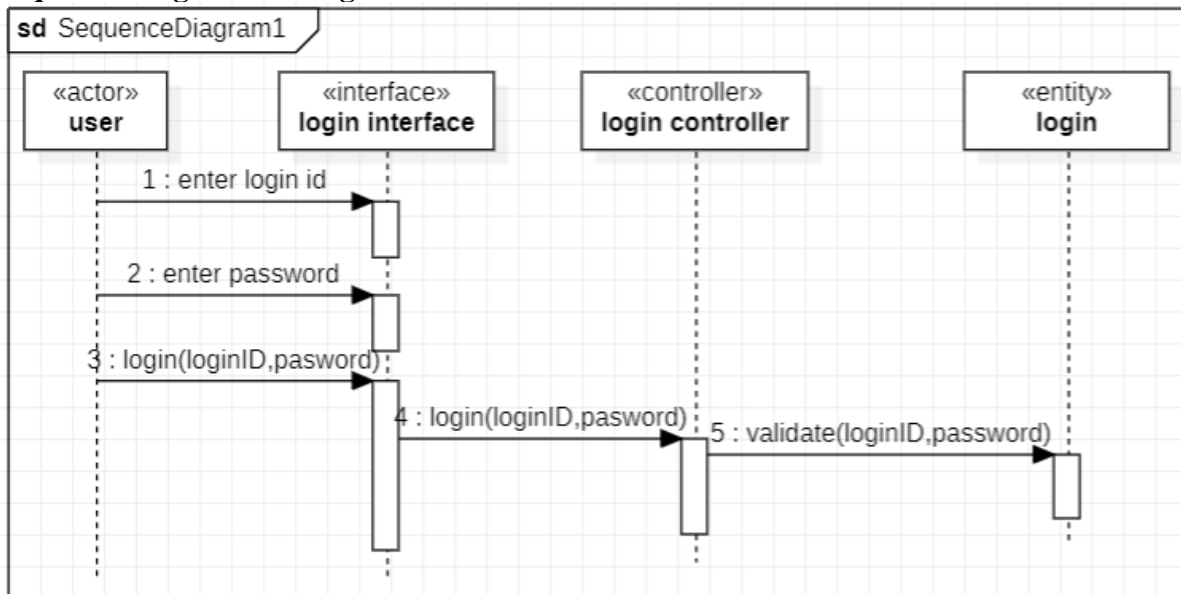
### Sequence diagram for Registration alternate flow 2



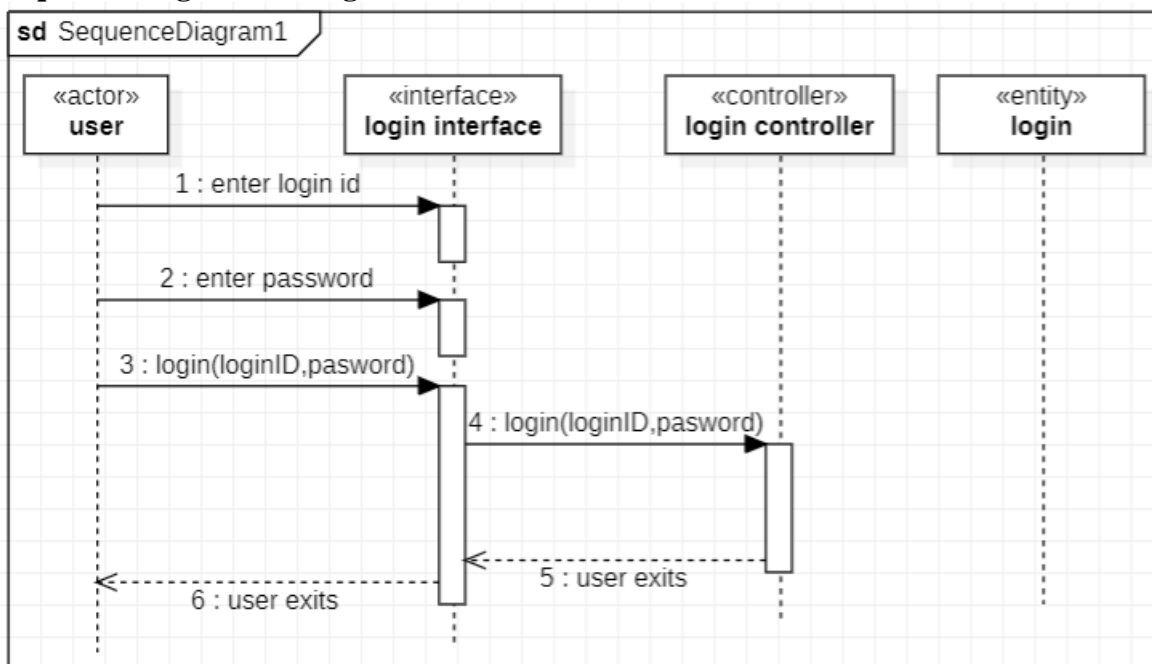
### Sequence diagram for Registration alternate flow 3



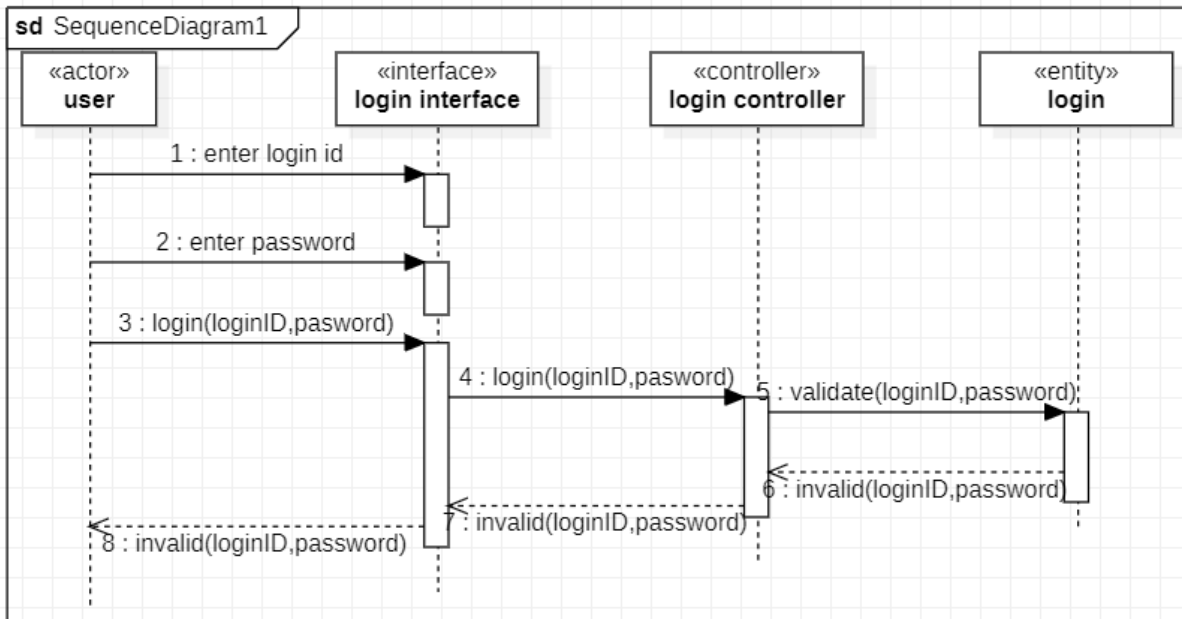
### Sequence diagram for Login basic flow 1



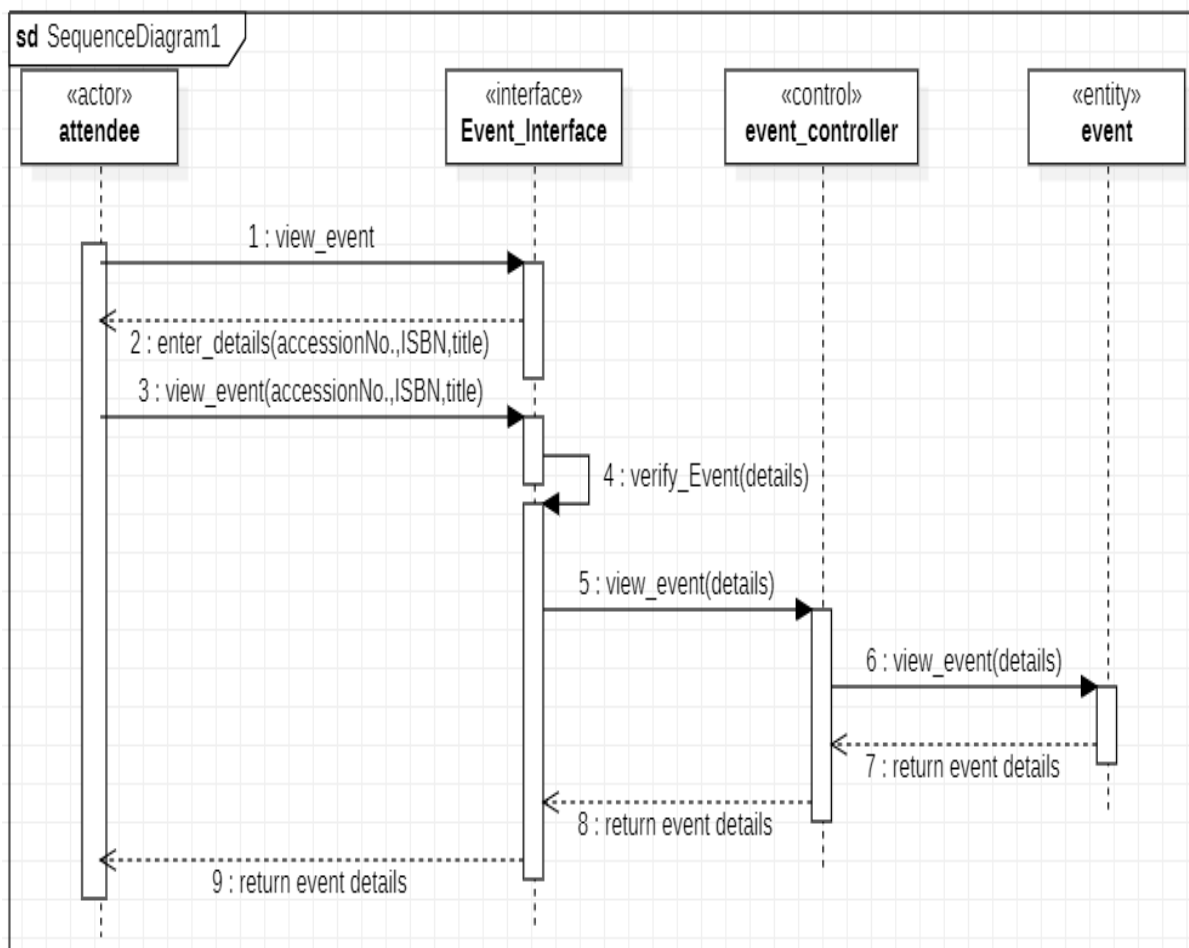
### Sequence diagram for Login alternate flow 1



### Sequence diagram for Login alternate flow 2

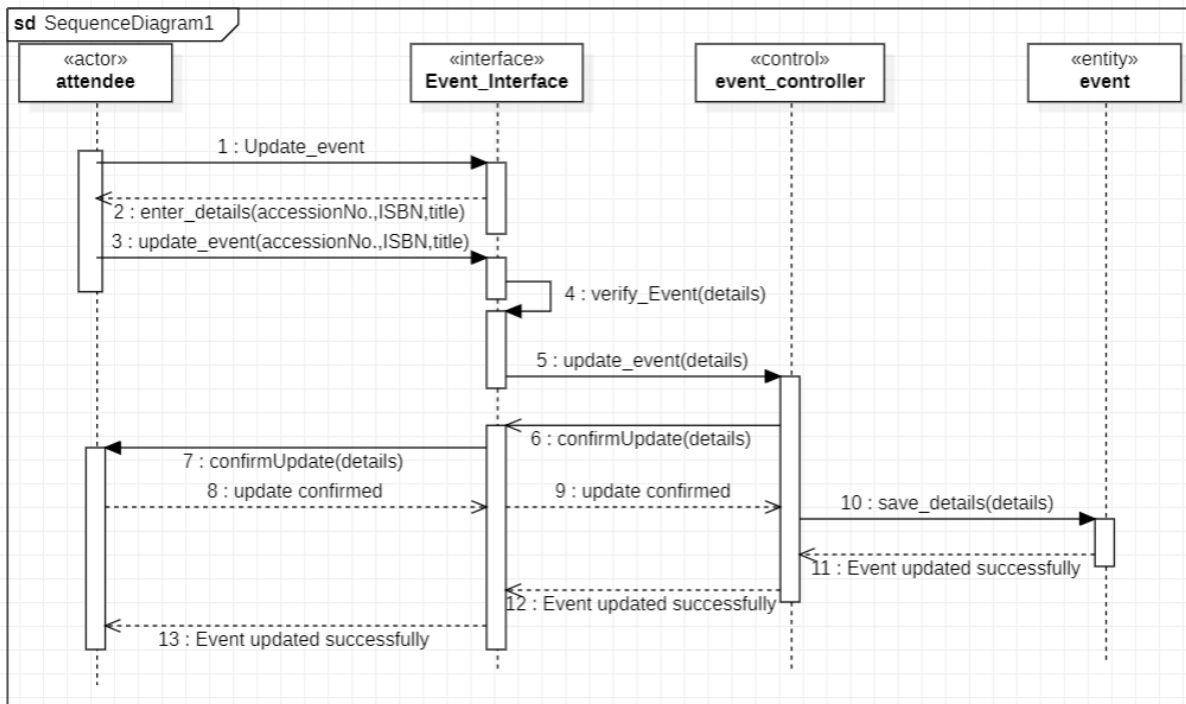


### Sequence diagram for View event basic flow

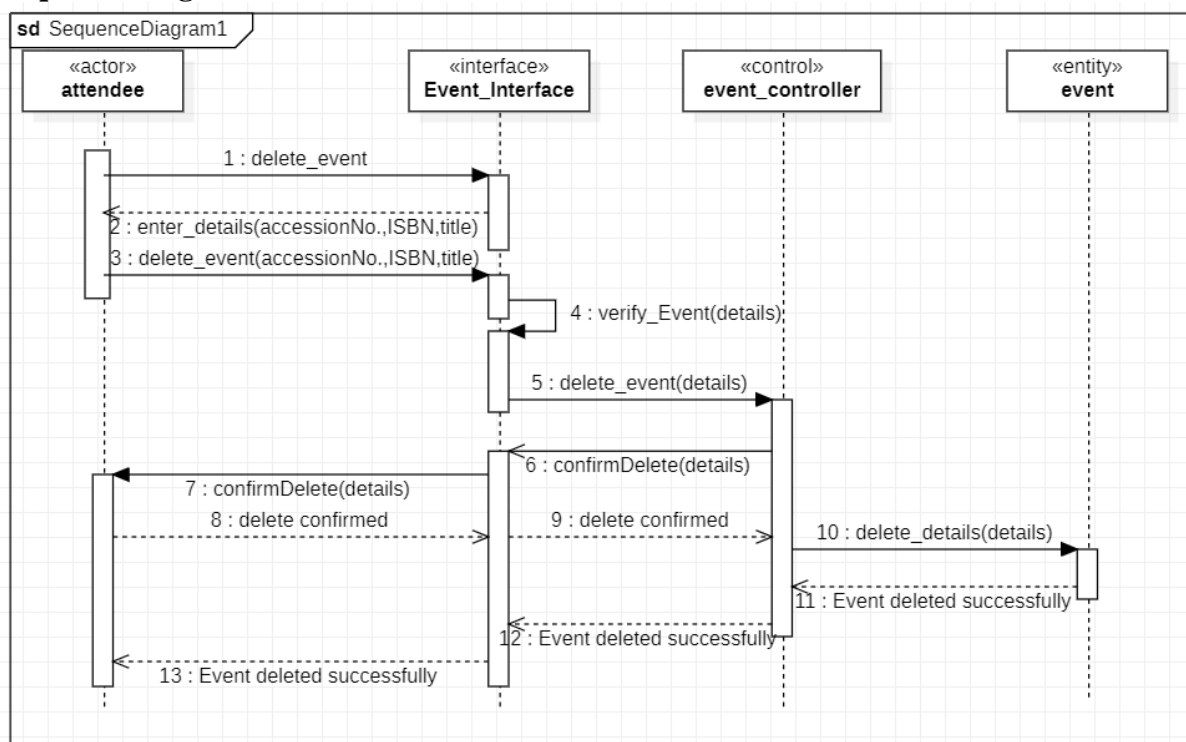




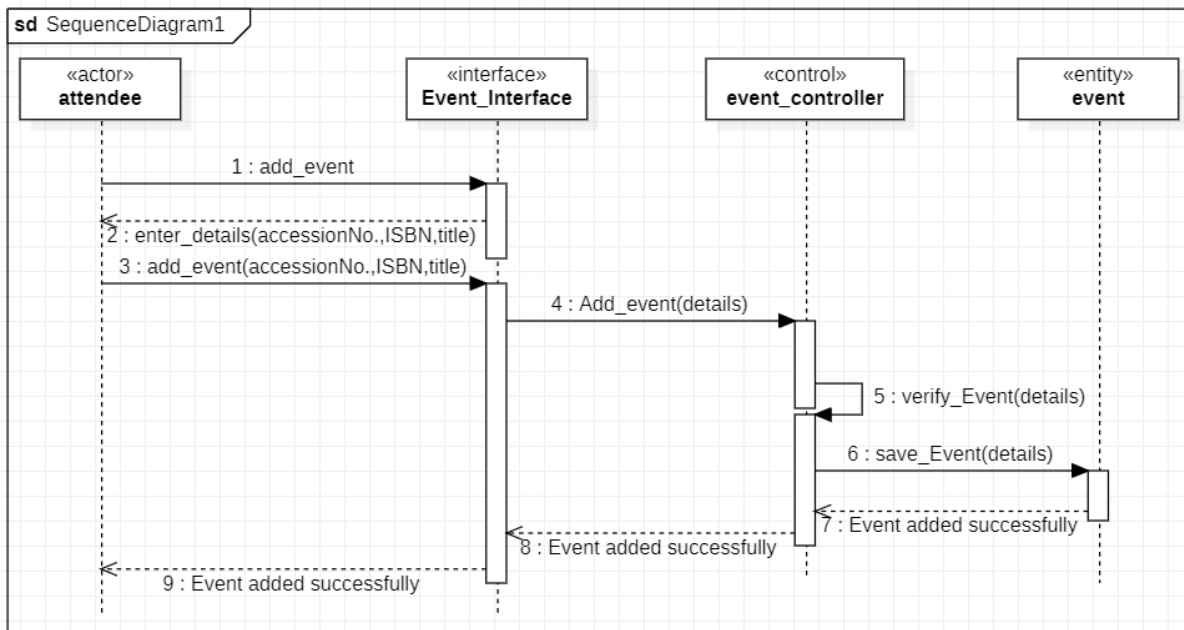
## Sequence diagram for Update event



## Sequence diagram for Delete event



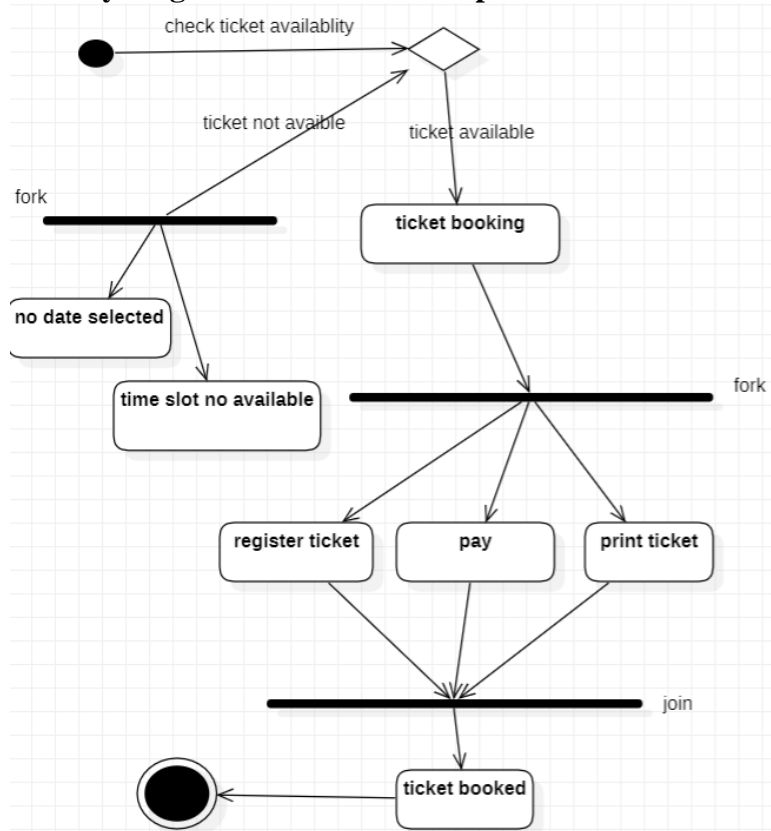
## Sequence diagram for Add event



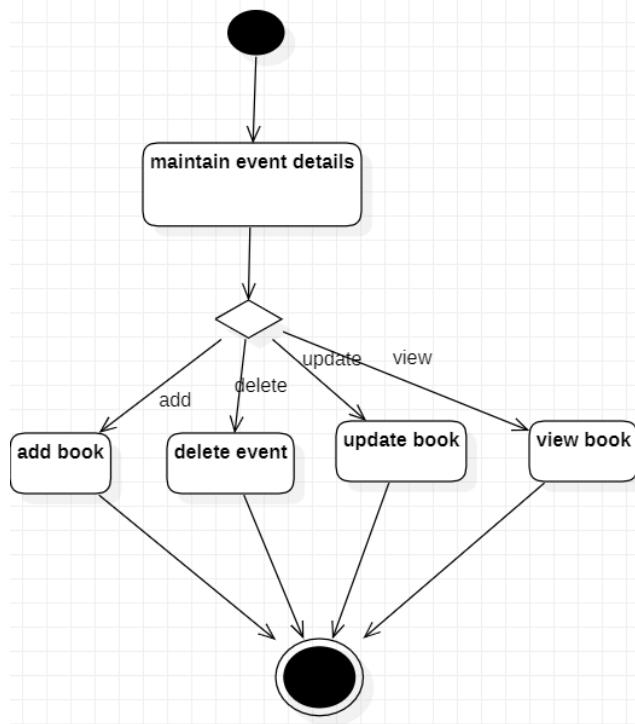
## **EXPERIMENT-8**

**AIM-** Make activity diagram for event website using star-uml.

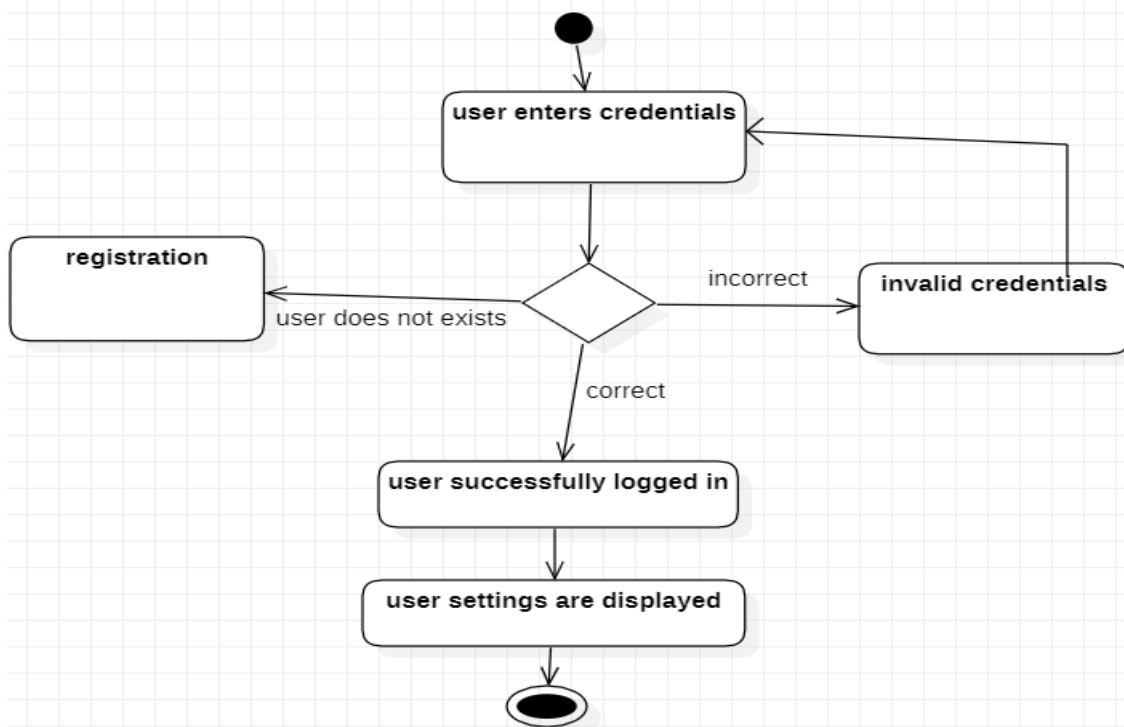
### **Activity diagram for Book tickets/passes**



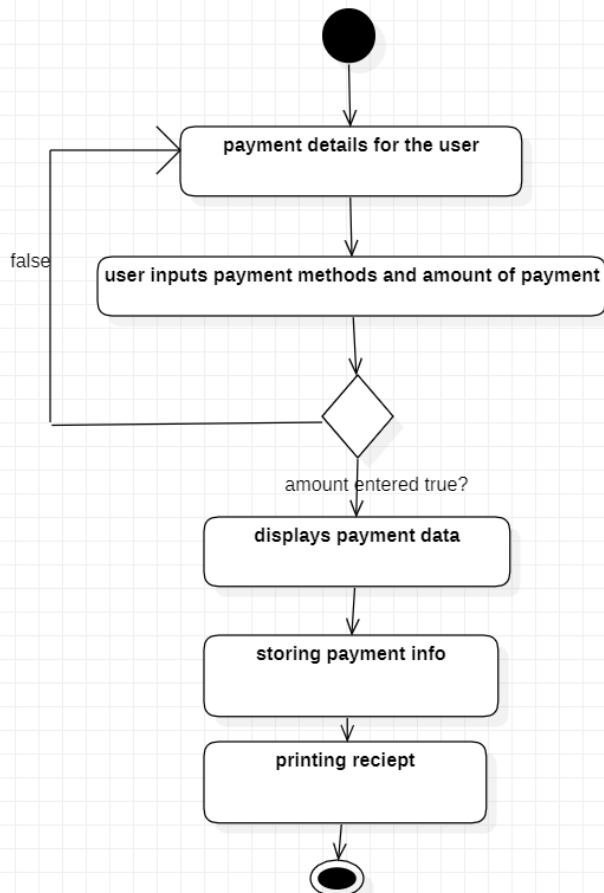
### **Activity diagram for Event details**



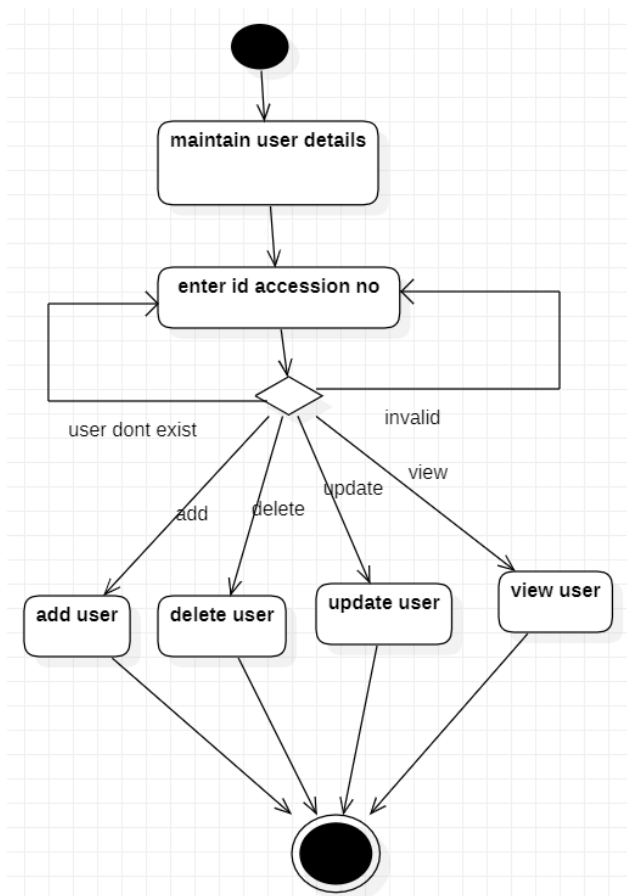
### Activity diagram for Login



### Activity diagram for Manage transactions



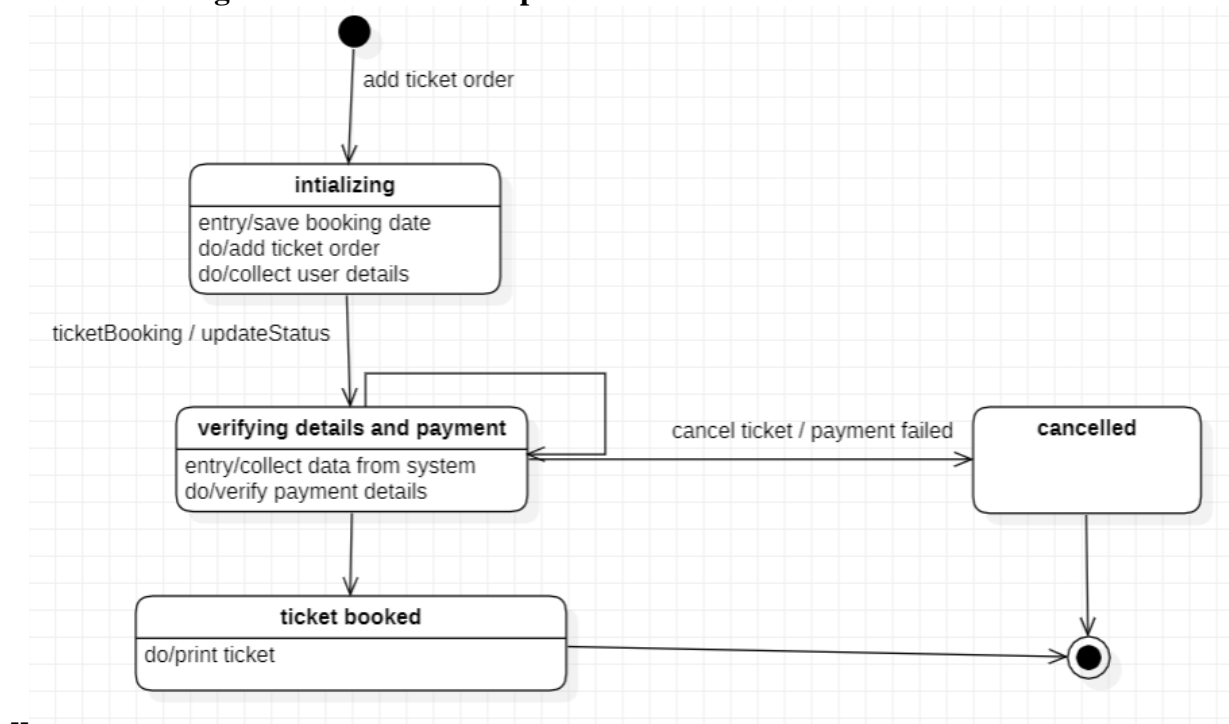
## Activity diagram for User details



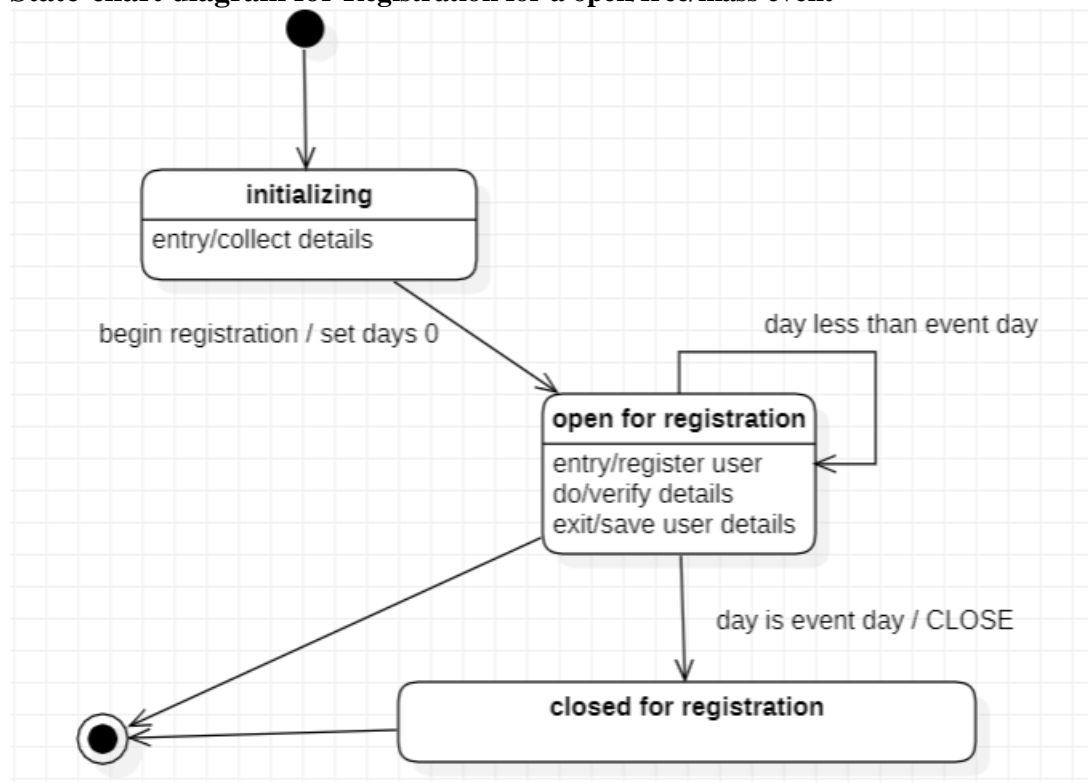
## EXPERIMENT-9

**AIM-** Make state-chart diagram for event website using star-uml.

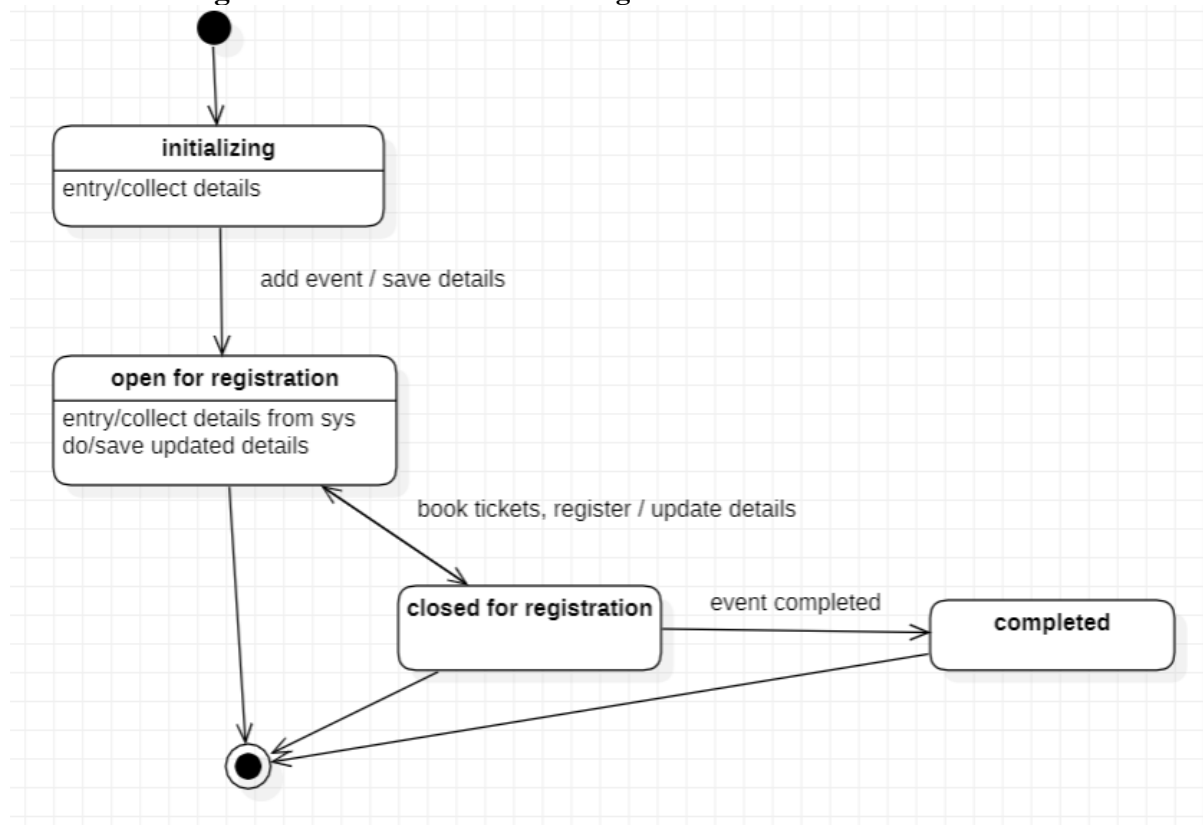
**State-chart diagram for Book tickets/passes**



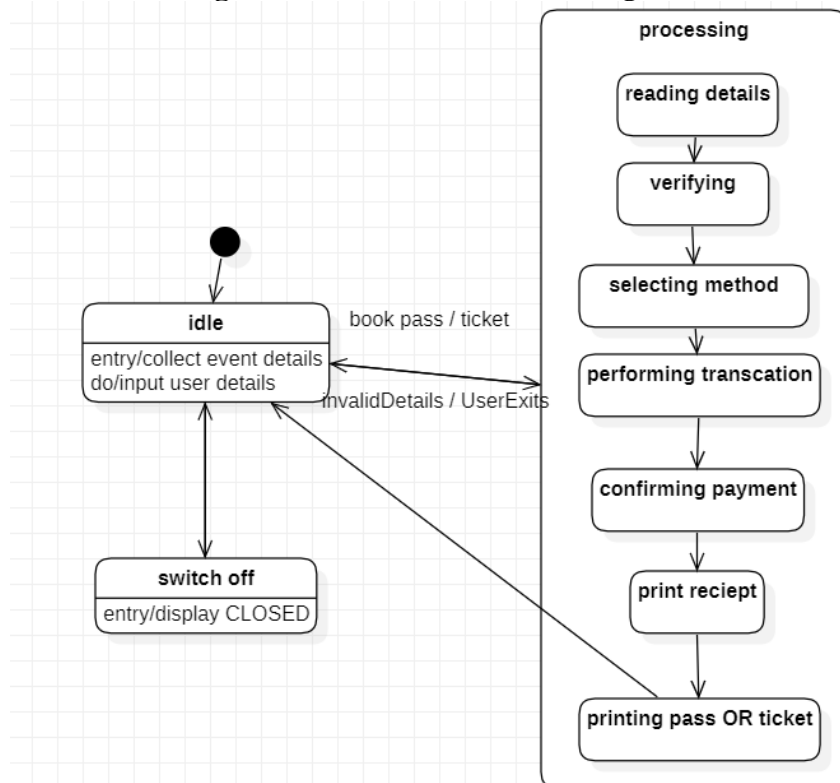
**State-chart diagram for Registration for a open/free/mass event**



### State-chart diagram for Event State Chart diagram



### State-chart diagram for Ticket State Chart diagram



## **EXPERIMENT-10**

**AIM-** Design test case matrixes for event website

### **Test case matrix for Issue Ticket use case**

<b>TEST CASE ID</b>	<b>SCENARIO NAME &amp; DESCRIPTION</b>	<b>EVENT ID</b>	<b>USER ID</b>	<b>EXPECTED ID</b>	<b>REMARKS</b>
TC1	Scenario 1- issue ticket basic flow	Valid input	Valid input	Ticket is issued succesfully	-
TC2	Scenario 1- issue ticket alternate flow: unauthorized event/user	Valid/invalid input	Valid/invalid input	User/event not registered	User/event is not validated by the system
TC3	Scenario 3— Issue ticket alternative flow: event is full	Valid input	Valid input	Event tickets are fully booked	Event is booked fully and no more tickets are available
TC4	Scenario 4— Issue ticket alternative flow:ticket is already reserved	Valid input	Valid input	Ticket is already booked from this account	Ticket is already purchased by the user form this account
TC5	Scenario 5- issue ticket alternate flow: Unable to fetch user details	Valid input	invalid input	Error in fetching user details	Details of the specified event is blank or not in readable form.
TC6	Scenario 6- issue ticket alternate flow: Unable to fetch event details	invalid input	Valid input	Error in fetching event details	Details of the specified event is blank or not in readable form.
TC7	Scenario 7 - issue ticket alternate flow: Payment failed	Valid input	Valid input	Error in payment	Payment is not done correctly or failed or balance is remaining



TC8	Scenario 8- issue ticket alternate flow : User exits	Valid input	Valid input	User comes out of the system	-
-----	---	-------------	-------------	------------------------------	---

**Test case matrix with actual data values for Issue Ticket use case**

TEST CASE ID	SCENARIO NAME & DESC	USER ID	EVENT ID	EXPECTED OUTPUT	REMARKS
TC1	Scenario 1- book ticket basic flow	1001	40001	Ticket is booked successfully	-
TC2	Scenario 2- issue ticket alternate flow: invalid User /event	1002	40002	User/event not registered	User/event is not validated by the system
TC3	Scenario 3— Issue ticket alternative flow: event is full	1000	40000	Event tickets are fully booked	Event is booked fully and no more tickets are available
TC4	Scenario 3- Issue ticket alternate flow: ticket already booked	1003	40003	Ticket to event is already registered	Ticket to event is already registered for the corresponding event.
TC5	Scenario 4- issue ticket alternate flow: Unable to fetch event details	1004	40004	Error in fetching event details	Details of the specified event is blank or not in readable form.
TC6	Scenario 5- issue ticket alternate flow: Unable to fetch user details	1005	40005	Error in fetching user details	Details of the specified event is blank or not in readable form.
TC7	Scenario 6- issue ticket alternate flow: Payment failed	1006	40006	Error in payment	Payment is not done correctly or failed or balance is remaining
TC8	Scenario 7- issue ticket alternate flow : User exits	1007	40007	User comes out of the system	-

### Test case matrix for the Login use case

TEST CASE ID	SCENARIO NAME AND DESC	LOGIN ID	PASSWORD	OLD PASSWORD	NEW PASSWORD	CONFIRM PASSWORD	EXPECTED OUTPUT	REMARKS (IF ANY)
TC1	Scenario 1—login	Valid input	Valid input	n/a	n/a	n/a	User is allowed to login	-
TC2	Scenario 2—login Alternate flow: Invalid Entry	Invalid input	Valid input	n/a	n/a	n/a	Login ID invalid	Login ID is not in specified format or incorrect
TC3	Scenario 2—login Alternate flow: Invalid Entry	Valid input	Valid input	n/a	n/a	n/a	Login ID invalid	Login ID does not exist in database.
TC4	Scenario 2—login Alternate flow: Invalid Entry	Valid input	Invalid input	n/a	n/a	n/a	Password invalid	Password is not in specified format which is less than 4 characters. OR incorrect
TC5	Scenario 2—login Alternate flow: Invalid Entry	Valid input	Valid input	n/a	n/a	n/a	Password invalid	Password does not exist in database.
TC6	Scenario 2—login Alternate flow: Invalid Entry	Invalid input	Invalid input	n/a	n/a	n/a	Login ID/password invalid	Login ID and password are not in the specified format. Login ID is less than 11 characters and password is less than 4 characters.
TC7	Scenario 3—Login alternative flow: User exit	Valid/invalid input	Valid/invalid input	n/a	n/a	n/a	User comes out of the system	-
TC8	Scenario 4—Change password	Valid input	n/a	Valid input	Valid input	Valid input	User is allowed to change password	Password is changed in the system
TC9	Scenario 5—Change password alternative flow: Invalid	Invalid input	n/a	Valid/invalid input	Valid/invalid input	Valid/invalid input	Old password is invalid	Login id is not correct

	entry							
TC10	Scenario 5— Change password alternative flow: Invalid entry	Valid input	n/a	Invalid input	Valid/invalid input	Valid/invalid input	Old password invalid	Old password does not match the corresponding password in the database. Other entries (new password and confirm password) become 'do not c a r e '.
TC11	Scenario 5— Change password alternative flow: Invalid entry	Valid input	n/a	Valid input	Invalid input	C	New password invalid	New password is not in the specified format which is less than 4 characters. Other entries (confirm password) become 'do not care'.
TC12	Scenario 5— Change password alternative flow: Invalid entry	Valid input	n/a	Valid input	Valid input	Valid input	Confirm password does not match new password	New and confirm password entries are different
TC13	Scenario 6— Change password alternative flow: User exits	Valid/invalid input	n/a	Valid/invalid input	Valid/invalid input	Valid/invalid input	User is allowed to exit and returns to login screen	-

### Test case matrix with actual data values for Login use case

TEST CASE ID	SCENARIO NAME AND DESC	LOGIN ID	PASSWORD	OLD PASSWORD	NEW PASSWORD	CONFIRM PASSWORD	EXPECTED OUTPUT	REMARKS (IF ANY)
TC1	Scenario 1—login	100007	dtu123	n/a	n/a	n/a	User is allowed to login	-
TC2	Scenario 2—login Alternate flow: Invalid Entry	458	dtu123	n/a	n/a	n/a	Login ID invalid	Login ID is not in specified format which is less than 6 characters
TC3	Scenario 2—login Alternate flow: Invalid Entry	100099	dtu123	n/a	n/a	n/a	Login ID invalid	Login ID does not exist in database or incorrect
TC4	Scenario 2—login Alternate flow: Invalid Entry	100007	R34	n/a	n/a	n/a	Password invalid	Password is not in specified format which is less than 4 characters.
TC5	Scenario 2—login Alternate flow: Invalid Entry	100007	dtu123	n/a	n/a	n/a	Login ID invalid	Password does not exist in database.
TC6	Scenario 2—login Alternate flow: Invalid Entry	9999	Bala7878787	n/a	n/a	n/a	Login ID/password invalid	Login ID and password are not in the specified format. Login ID is less than 11 characters and password is less than 4 characters.
TC7	Scenario 3—Logial ternative flow: User exit	*	*	n/a	n/a	n/a	User comes out of the system	-
TC8	Scenario 4—Change password	100007	n/a	dtu123	Dtu999	Dtu999	User is allowed to change password	-

TC9	Scenario 5— Change password alternative flow: Invalid entry	911	n/a	*	*	*	Login ID is invalid	Login ID is not in the specified format.
TC10	Scenario 5— Change password alternative flow:Invalid entry	100007	n/a	dtu123	*	*	Old password invalid	Old password does not match the correspondi ng password in the database. Other entries (new password and confirm password) become ‘do not c a r e’.
TC11	Scenario 5— Change password alternative flow: Invalid entry	100007	n/a	dtu123	Bala778788	*	New password invalid	New password is not in the specified format which is less than 4 characters. Other entries (confirm password) become ‘do not care’.
TC12	Scenario 5— Change password alternative flow: Invalid entry	100007	n/a	dtu123	dtu123	dtu123	Confirm password does not match new password	-
TC13	Scenario 6— Change password alternative flow:User exits	*	n/a	*	*	*	User is allowed to exit and returns to login screen	-