**Use case description for registration**

|  |  |
| --- | --- |
|  | **Brief description:** This use case allows an actor to register securely in the system  Creating and saving data in the database |
|  | **Actor:** attendee and admin |
|  | **Precondition:** none |
|  | **Postcondition:** the attendee is successfully registered and his details are securely saved |
|  | **Basic flow:**   1. The use case starts when the user clicks the “Sign Up” tab on the navigation bar. 2. The system displays the signup page that allow users to fill in their **username/email address**, **first** and **last name** and their **password**. 3. The user must enter a valid username that does not already exists. 4. The user keys in the details and clicks “Sign Up” button. 5. The system displays the user’s profile page. |
|  | **Alternate flow:**   1. **Invalid details-** 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. 2. **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. |
|  | **Special requirements:** N/A |
|  | **Associated use case:** N/A |

**use case description for login**

|  |  |
| --- | --- |
|  | **Brief description:** user tries to sign into the system to view there details. |
|  | **Actor:** attendee and admin |
|  | **Precondition:** user should be registered before hand in the database |
|  | **Postcondition:** the user has access to the system |
|  | **Basic flow:**   1. The use case starts when the user clicks the “Login” tab on the navigation bar 2. The system displays the login page that allow users to fill in their username/email address and password 3. The user keys in the details and clicks “Sign In” button 4. The system checks the details in the database 5. The system displays the user’s profile page |
|  | **Alternate flow:**   1. **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. 2. **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. |
|  | **Special requirements:** N/A |
|  | **Associated use case:** N/A |

**use case description for maintain user details**

|  |  |
| --- | --- |
|  | **Brief description:** 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 |
|  | **Actor:** admin |
|  | **Precondition:** admin must be logged into the system before this use case begins |
|  | **Postcondition:** if the use case is successful, then user information is added, updated , deleted or viewed. |
|  | **Basic flow:**   1. The use case starts when the admin wished to add/update/delete/view user information. 2. Once the admin provides the requested information, one of the following subflows is executed-  * if the admin selects “ add user ”, the add a user subflow is executed * if the admin selects “update user details ”,the update user details subflow is executed * if the admin selects “delete user details ”,the delete user details subflow is executed * if the admin selects “view user details ”,the view user details subflow is executed   **basic flow 1(add a user)**  the system requests that the admin enter the user details.this includes**-**   1. **a** |
|  | **Alternate flow:**   1. **invalid entry-** 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. 2. **User not found-** 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. 3. **Deletion not allowed-**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 |
|  | **Special requirements:** N/A |
|  | **Associated use case:** N/A |

**use case description for event details**

|  |  |
| --- | --- |
|  | **Brief description:** this use case provides all the details about the events |
|  | **Actor:** attendee, admin and event organiser |
|  | **Precondition:** user should be logged into the system and must provide his/her location |
|  | **Postcondition:** the user has access to all information about the events and location details. |
|  | **Basic flow:**   1. The use case starts when the user clicks the “events details” 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 events on the basis of his/her location |
|  | **Alternate flow:**   1. **Login failed**-if the user is not logged into the system the the system will show and error and redirects to the login page in order to login again. 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. |
|  | **Special requirements:** N/A |
|  | **Associated use case:** N/A |

**use case description for book tickets**

|  |  |
| --- | --- |
|  | **Brief description:** this use case provides all the details about the tickets and purchase tickets for the corresponding events. |
|  | **Actor:** attendee, admin and event organiser |
|  | **Precondition:** user should be logged into the system and must provide his/her location |
|  | **Postcondition:** the user has access to all information about the events and location details. |
|  | **Basic flow:**   1. The use case starts when the user clicks the “events details” 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 events on the basis of his/her location |
|  | **Alternate flow:**   1. **Login failed**-if the user is not logged into the system the the system will show and error and redirects to the login page in order to login again. 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. |
|  | **Special requirements:** N/A |
|  | **Associated use case:** |