Use Cases (Group 4)

|  |  |
| --- | --- |
| **Name** | Administrator – Add Facility. |
| **ID** | UC\_001 |
| **Description** | An Admin wants to add a facility to a specific school. |
| **Actors** | School Administrator. |
| **Organizational Benefits** | By adding a facility to a specific school’s page, they will be able to advertise the use of this facility, inform the public about upcoming events being hosted there, and allow people to see pictures and information about the facility. |
| **Triggers** | The Admin selects the “Add Facility” option. |
| **Preconditions** | The user who wants to add the facility must be an Admin of a specific school. |
| **Postconditions** | The facility and other associated information is stored for a specific school and is accessible to other users. |
| **Main Course** | 1. Admin clicks “Add Facility”. 2. Admin uploads pictures, maximum capacity, name of facility, and location. 3. Admin clicks “Save Facility”. 4. System confirms facility has been added. 5. System redirects to admin homepage. |
| **Alternate course** | 1. User is not an Admin. 2. Connection times out. 3. No new information is added to the facility. |
| **Exceptions** | 1. Facility already exists. |

|  |  |
| --- | --- |
| **Name** | Login. |
| **ID** | UC\_002 |
| **Description** | A user wants to login to their account |
| **Actors** | User |
| **Organizational Benefits** | Users can create accounts which allows them to purchase tickets to events, donate money to schools, and see events being hosted at facilities near them. |
| **Triggers** | User clicks “Login”. |
| **Preconditions** | The user is not currently logged in as a user. |
| **Postconditions** | The user is now logged in to their account. |
| **Main Course** | 1. User clicks “Login” |
| **Alternate Course** | 1. User wants to buy tickets but isn’t logged in. 2. User wants to donate but isn’t logged in. 3. Connection timed out. 4. Admin wants to add a facility but isn’t logged in. 5. User wants to request / add event but isn’t logged in. 6. User wants to share an event but isn’t logged in. 7. Admin wants to edit a facility but isn’t logged in. |
| **Exceptions** | 1. The user is not logged in as a user. |

|  |  |
| --- | --- |
| **Name** | Create coordinator user. |
| **ID** | UC\_003 |
| **Description** | The school administrators want a coordinator user to manage the facilities that are registered and the events being held in them. |
| **Actors** | Coordinator. |
| **Organizational Benefits** | Allows limited access to facility and event information and approvals. |
| **Triggers** | The coordinator selects the create coordinator user option. |
| **Preconditions** | The user enters a unique access code that gives them permission to create a coordinator account. |
| **Postconditions** | The coordinator user is created. |
| **Main Course** | 1. System prompts user to enter access code. 2. User enters necessary information to create the account. 3. User submits information when finished. 4. System confirms that account has been created. 5. System redirects user to login page. |
| **Alternate Course** | 1. System does not accept access code. |
| **Exceptions** | 1. User is already logged in. |

|  |  |
| --- | --- |
| **Name** | Search for an event. |
| **ID** | UC\_004 |
| **Description** | Users are able to search for pre-existing events in the database. |
| **Actors** | All users and non-users (not logged in). |
| **Organizational Benefits** | Generates potential income by allowing for events to be searched. |
| **Triggers** | The user selects the option to search for an event. |
| **Preconditions** | User enters desired search criteria. |
| **Postconditions** | Search returns results based on search criteria. |
| **Main Course** | 1. System prompts user for search criteria. 2. User enters desired information. 3. User submits information. 4. System reads information. 5. System returns results. |
| **Alternate Course** | 1. User has not completed all fields in search form. 2. User has illegal characters in any given search field. 3. User navigates away from search page. |
| **Exceptions** | 1. System does not return any results. 2. System redirects to another page. 3. System truncates and/or changes part of the search criteria. |

|  |  |
| --- | --- |
| **Name** | Request event. |
| **ID** | UC\_005 |
| **Description** | Users are able to request to host an event in an available facility. |
| **Actors** | All users. |
| **Organizational Benefits** | Generate revenue by allowing the space to be rented for events. |
| **Triggers** | The user selects the option that they wish to request an event. |
| **Preconditions** | User must be signed in. |
| **Postconditions** | The event request is sent. |
| **Main Course** | 1. User logs in successfully. 2. System prompts user to select the facility. 3. User selects facility. 4. System prompts user to select available timeframe. 5. User selects. 6. System prompts user to fill out form regarding the event/activity. 7. User fills out form. 8. User submits form. |
| **Alternate Course** | 1. User does not fill out form all the way. 2. The event has already been requested. 3. User has illegal characters in any field. 4. User navigates away from request page. |
| **Exceptions** | 1. The user is no longer signed in. 2. User selects an unavailable facility. 3. User selects an unavailable time frame. 4. The request is not successfully sent or does not bring up confirmation page . 5. System times out for request. 6. Request for available facility and timeframe is no longer available (if one user submits their request before another user is able to submit their request). |

|  |  |
| --- | --- |
| **Name** | Purchase tickets. |
| **ID** | UC\_006 |
| **Description** | User wants to purchase tickets for an event. |
| **Actors** | User, Billing system. |
| **Organizational Benefits** | Generate revenue by selling tickets. |
| **Triggers** | The user selects the purchase tickets option and the number of tickets they want. |
| **Preconditions** | User selects the event they want to purchase tickets for. |
| **Postconditions** | The order will be placed in the system and the user will get a confirmation that the order was successful. |
| **Main Course** | 1. User indicates they want to place an order for the selected tickets. 2. System prompts the user to enter the billing and shipping information. 3. System displays the total cost of the order, including applicable taxes and shipping charges. 4. User confirms that the order information is accurate. 5. System submits the order to the fulfillment system for evaluation. 6. User places the order. 7. System requests that the billing system should charge for the order. 8. System indicates confirmation for placing the order and applying charges. 9. User exits the system. |
| **Alternate Course** | 1. Tickets are sold-out. |
| **Exceptions** | 1. System directs user to another event or facility. |

|  |  |
| --- | --- |
| **Name** | Donate to an event/facility. |
| **ID** | UC\_007 |
| **Description** | User wants to donate to an event or a facility. |
| **Actors** | User, Billing System. |
| **Organizational Benefits** | Generate more revenue by accepting donations. |
| **Triggers** | The user indicates they want to donate to an event or a facility by selecting donation option. |
| **Preconditions** | User selects the donation option and the amount to donate. |
| **Postconditions** | The donation is placed and accepted. User receives a donation confirmation and a “thank you” message. |
| **Main Course** | 1. User indicates they want to donate to an event or facility. 2. System prompts the user to enter the amount they wish to donate. 3. System prompts the user to enter their billing information. 4. User confirms that the donation information is accurate. 5. System submits the donation order to the fulfillment system for evaluation. 6. Billing system charges the donation that was placed. 7. System indicates donation was successful and user has been charged. |
| **Alternate Course** | 1. The event/facility is not accepting donations at that time. |
| **Exceptions** | 1. User is redirected to a donation option of another event. |

|  |  |
| --- | --- |
| **Name** | Share Event. |
| **ID** | UC\_008 |
| **Description** | Users are able to invite and share events with other people. |
| **Actors** | All users. |
| **Organizational Benefits** | Allows more people to know about the events being hosted. |
| **Triggers** | The user selects the share event option. |
| **Preconditions** | The event must exist. |
| **Postconditions** | The event is shared or a person is invited. |
| **Main Course** | 1. User logs in. 2. User selects an event they wish to share. 3. Event is shared to their page. 4. The user logs in. 5. The user selects an event. 6. The user selects invite. 7. User selects people they wish to invite. 8. Invites are sent to the other users. |
| **Alternate Course** | 1. The invited user doesn’t exist. 2. The event doesn’t exist. 3. Event is over capacity. 4. Event has already passed. 5. Event has been cancelled. |
| **Exceptions** | 1. Invite was not able to send. 2. Event could not be shared. |

|  |  |
| --- | --- |
| **Name** | Add Event. |
| **ID** | UC\_009 |
| **Description** | Users will be able to create events given an open facility. |
| **Actors** | All users. |
| **Organizational Benefits** | Generate revenue from ticket sales. |
| **Triggers** | User clicks the add event option. |
| **Preconditions** | Facility must be available during desired date and time. |
| **Postconditions** | Event is created and scheduled. |
| **Main Course** | 1. User clicks create event. 2. System displays list of available facilities. 3. User selects facility from list. 4. User enters necessary event information. 5. System prompts for date and time. 6. User sets date and time. 7. System displays list of terms and conditions. 8. User accepts. |
| **Alternate Course** | 1. Facility is in use. 2. Facility is undergoing repairs. 3. Terms and conditions were not accepted. 4. Event with same name already exists. |
| **Exceptions** | 1. System request times out. 2. System allows a facility to be booked twice. 3. System gives no terms and conditions. |

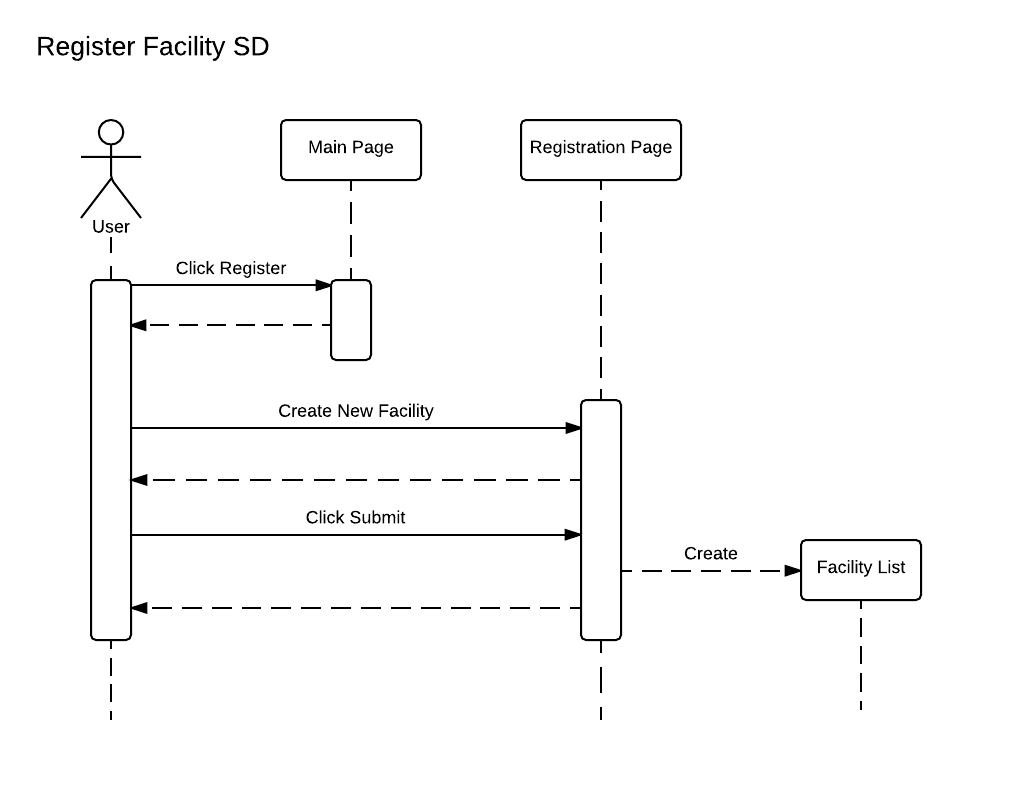
|  |  |
| --- | --- |
| **Name** | Manage Facility |
| **ID** | UC\_010 |
| **Description** | User will be able to add, delete and update |
| **Actors** | Coodinator. |
| **Organizational Benefits** | By adding, deleting and updating it will allow the system to be up to date. |
| **Triggers** | Click on add, delete or update. |
| **Preconditions** | The user who wants to add, delete and update the facility must be an Admin. |
| **Postconditions** | All the information stored must be accessed by all users. |
| **Main Course** | 1. User clicks the add option to add a new facility. 2. User clicks the delete option to delete an existing facility. 3. User clicks the update option to update an existing facility. |
| **Alternate Course** | 1. Admin clicks “Add, delete or update Facility”. 2. Admin, name of facility, dates and location. 3. Admin clicks “Save updates”. 4. System confirms facility has been added, deleted and updated. 5. System redirects to admin homepage. |
| **Exceptions** | 1. User is not an Admin. 2. Already updated. |

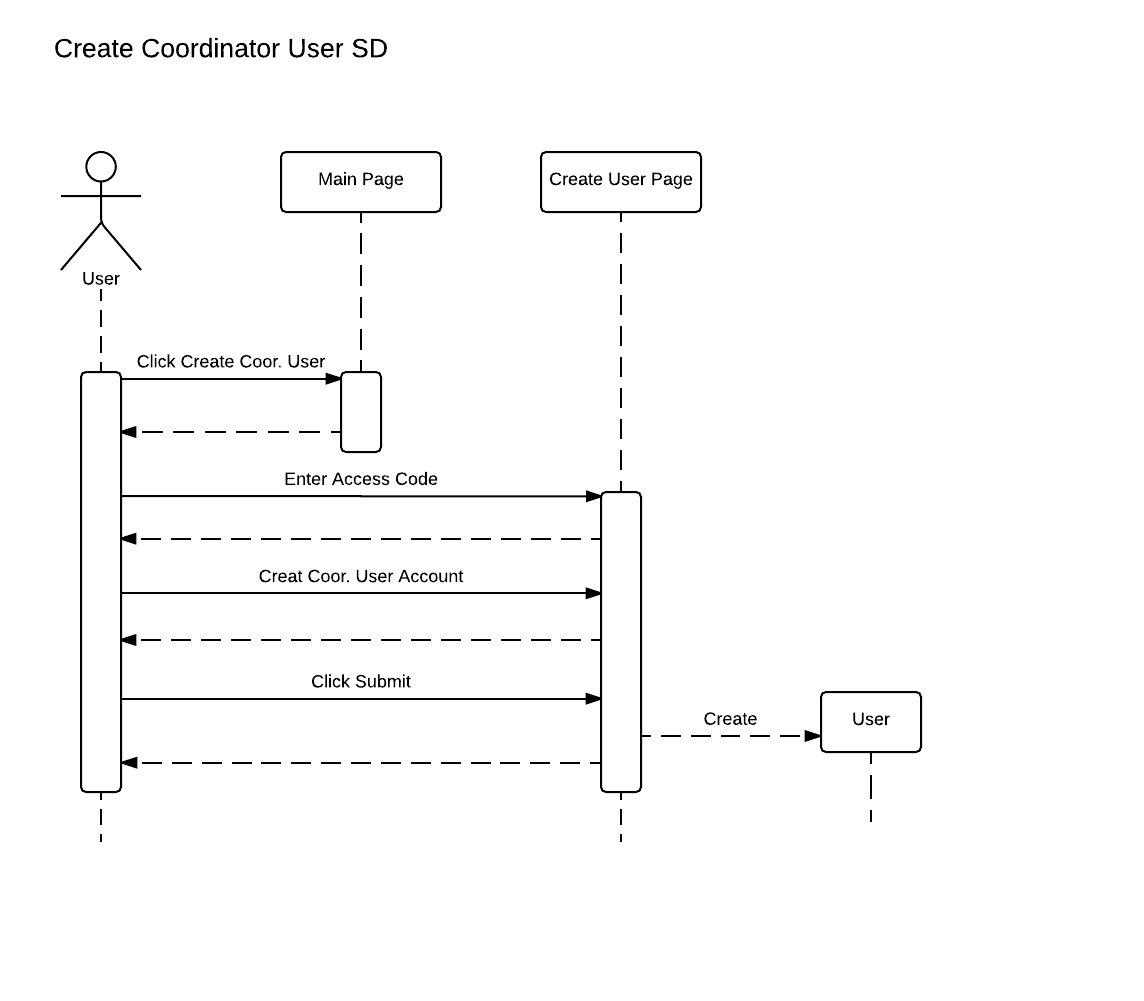
|  |  |
| --- | --- |
| **Name** | Export Data. |
| **ID** | UC\_011 |
| **Description** | User will be able to send data. |
| **Actors** | Coodinator. |
| **Organizational Benefits** | Keeping the data updated. |
| **Triggers** | Click “Send data” on the home page. |
| **Preconditions** | The data must be updated. |
| **Postconditions** | The data is stored. |
| **Main Course** | 1. User selects the send data option. 2. Add, send or update the data. |
| **Alternate Course** | 1. Admin click on “Add, Send Data”.   2. Admin click on update  3. Admin clicks “Save updates”. |
| **Exceptions** | 1. User is not the Admin. |

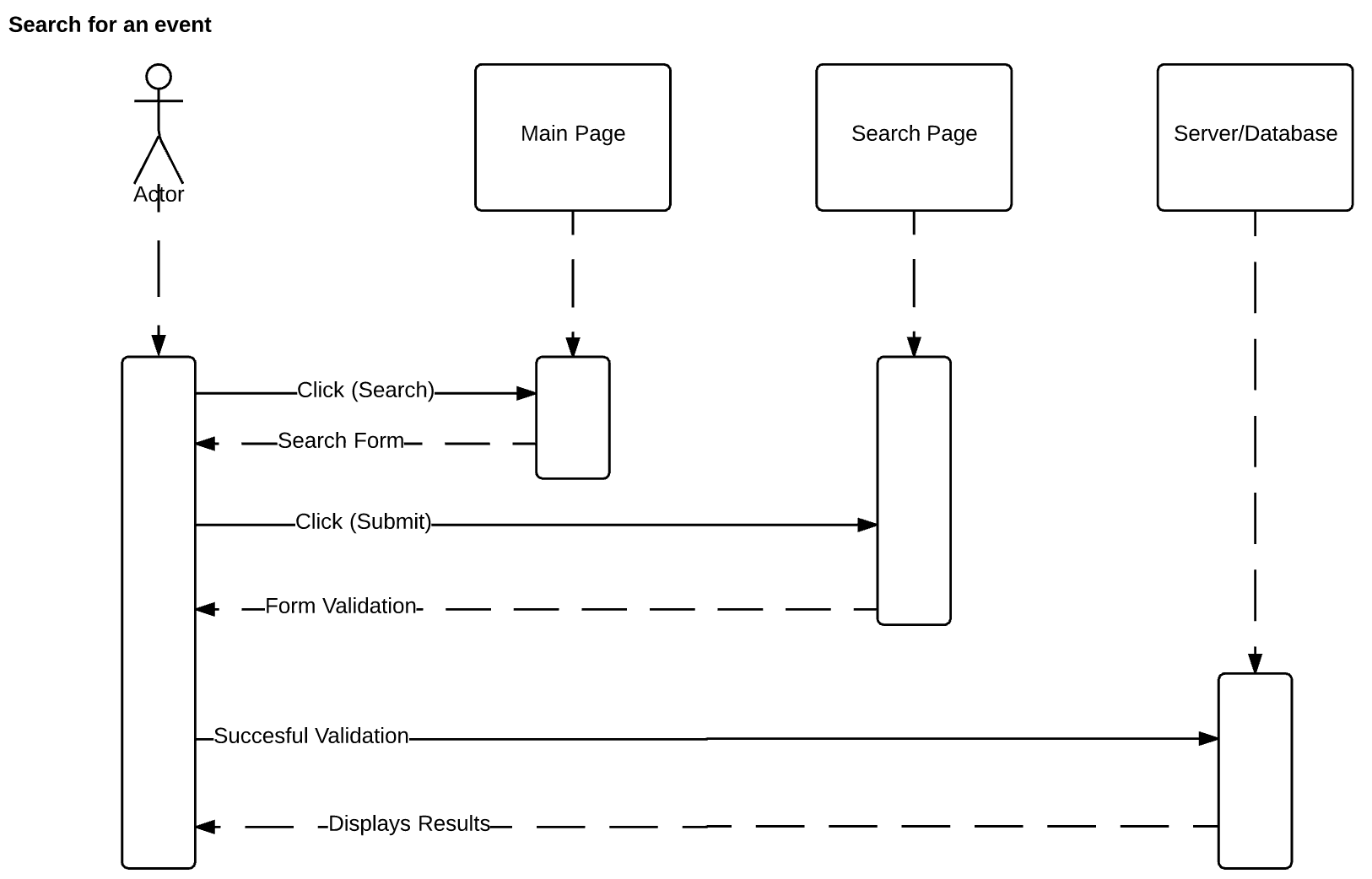
|  |  |
| --- | --- |
| **Name** | Edit Facility. |
| **ID** | UC\_012 |
| **Description** | Edit the details on a facility. |
| **Actors** | Coordinator. |
| **Organizational Benefits** | Display correct availability information to public. |
| **Triggers** | Coordinator selects the edit option on their Facility's page. |
| **Preconditions** | Coordinator must be logged in. Coordinator must select facility it has privileges to. |
| **Postconditions** | The Facilities page is changed based on Coordinator’s input. |
| **Main Course** | 1. The Coordinator clicks edit. 2. System displays editable fields. 3. Coordinator inputs desired values and presses confirm. 4. System saves and updates facility information. 5. System sends a confirmation message to Coordinator. |
| **Alternate Course** | 1. Facility does not exist. 2. Coordinator enters invalid information. |
| **Exceptions** | 1. Facility changes are not successful. |

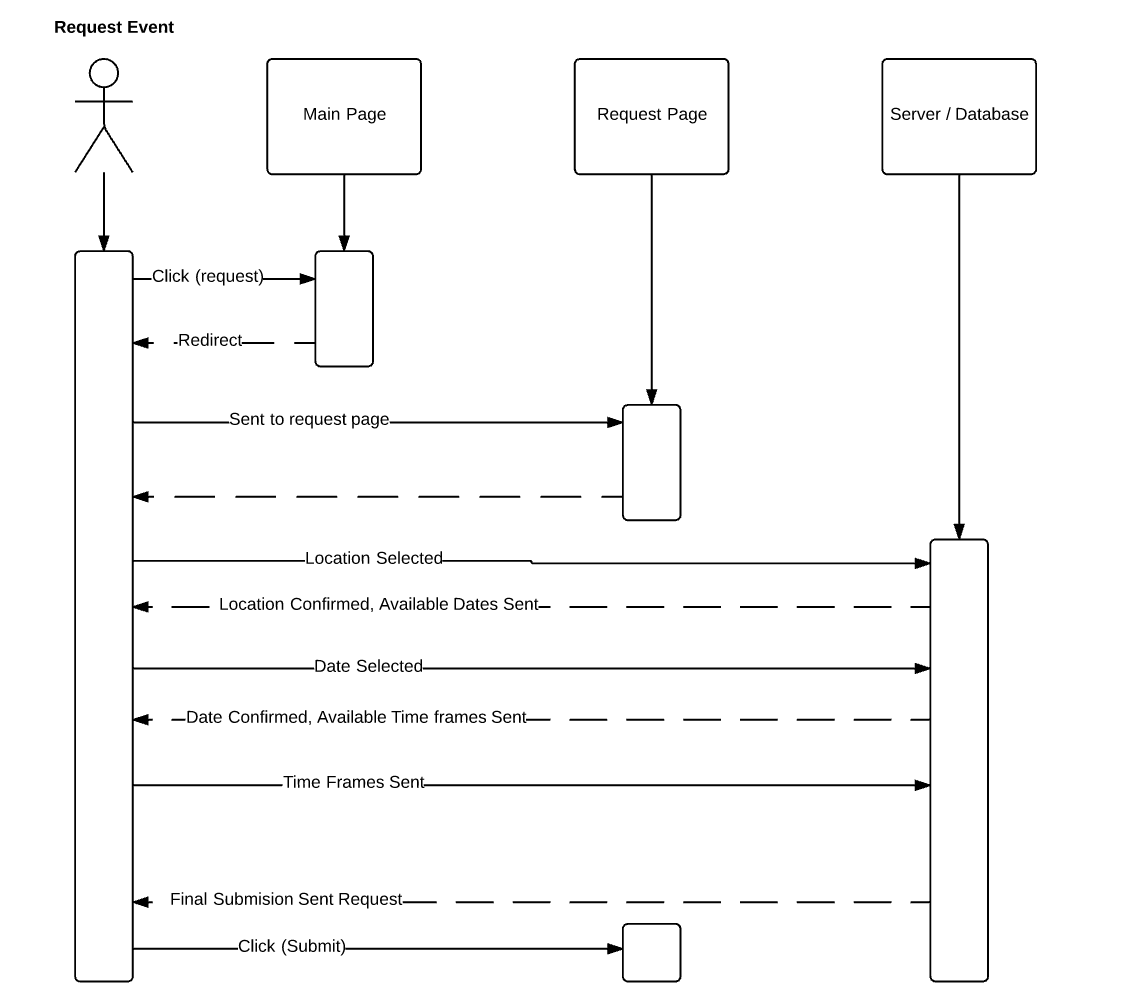
|  |  |
| --- | --- |
| **Name** | Refund Donation. |
| **ID** | UC\_013 |
| **Description** | Administrator refunds a user’s donation. |
| **Actors** | Administrator, User. |
| **Organizational Benefits** | Allows for the easy reversing of a transaction. |
| **Triggers** | Administrator clicks the refund option after fulfilling the appropriate fields. |
| **Preconditions** | Administrator must identify the user and transaction.  Administrator must identify the reason for refund. |
| **Postconditions** | The specified transaction should be reversed and notifications sent. |
| **Main Course** | 1. Administrator clicks the refund option after fulfilling the appropriate fields. 2. System authorizes a refund. 3. System updates appropriate data. 4. Notifications are sent to the Administrator and User. |
| **Alternate Course** | 1. Refund is not successful. |
| **Exceptions** | 1. There is no refund available. |

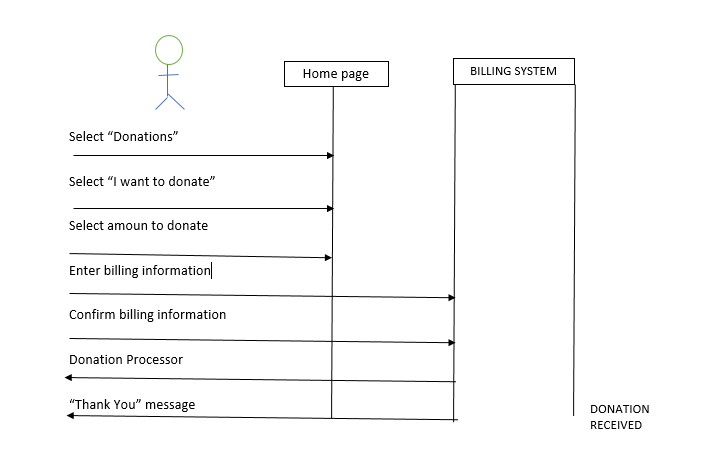
Sequence Diagrams

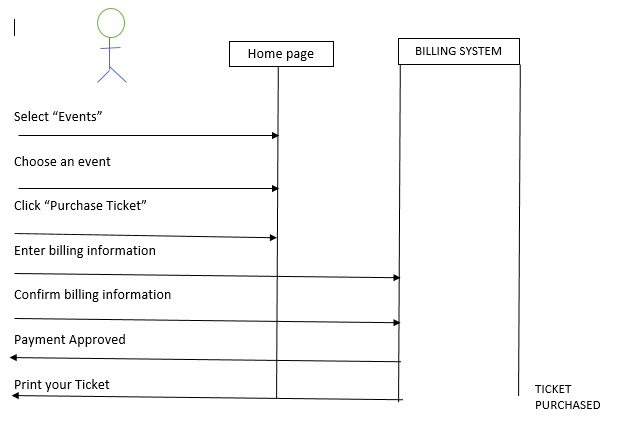


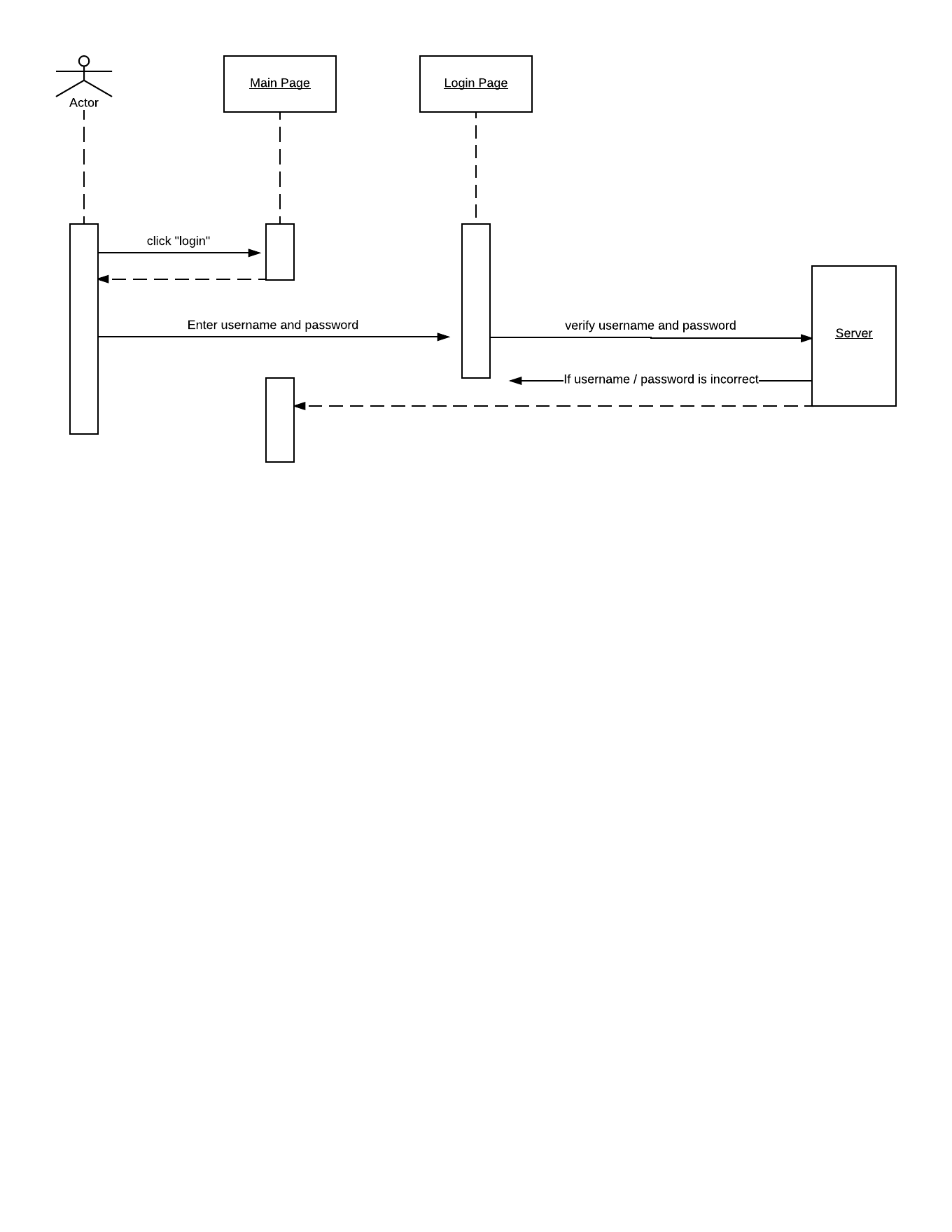






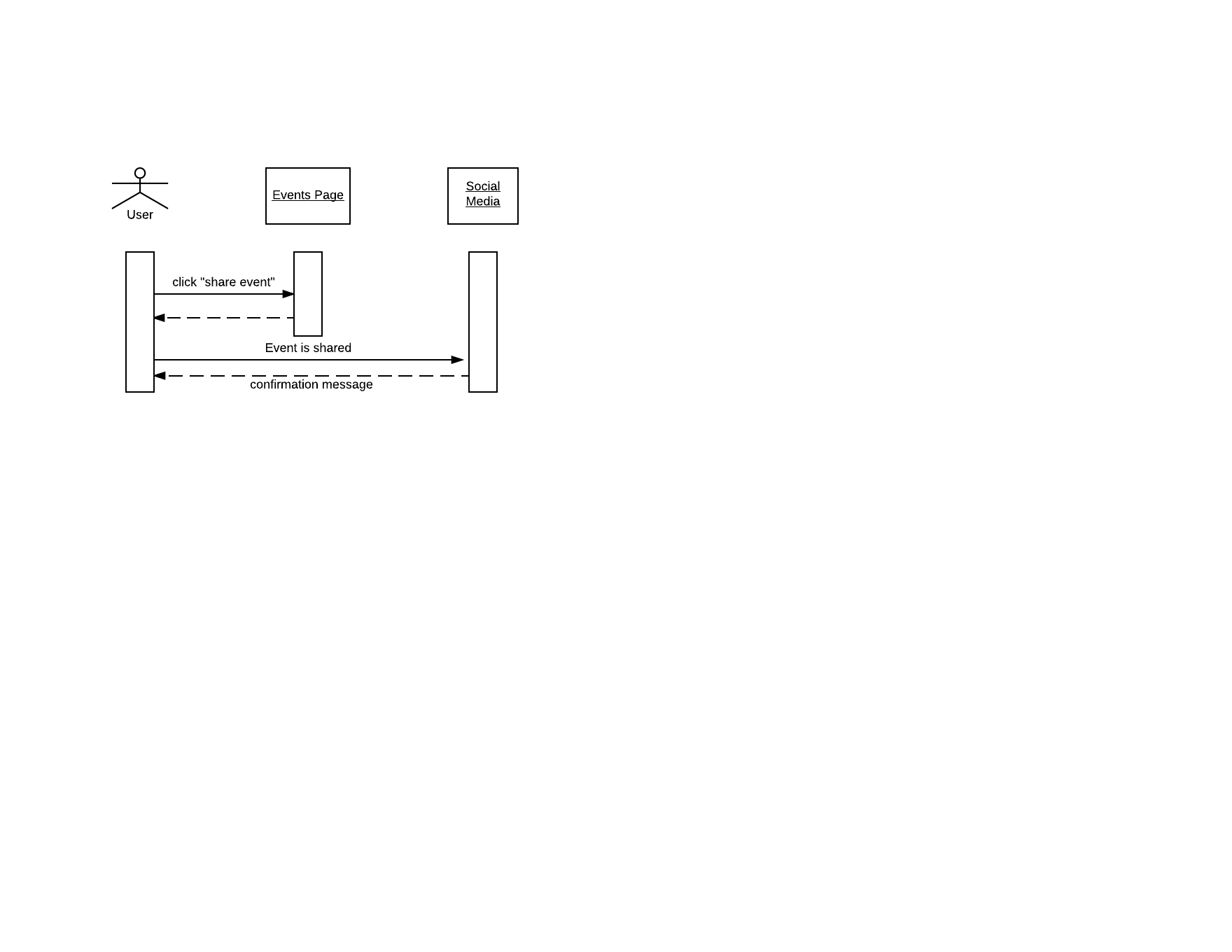






Login UC\_02

Share Event UC\_09



Add Event UC\_10

