# Design Use Cases

## Team SUS



| Trish Lam | Project Manager |
|---|---|
| Daniel Xiong | Business Analyst (Deputy #1) |
| Brian Nguyen | Senior System Analyst |
| Mary Nguyen | Software Architect |
| Victor Ung | Software Development Lead |
| Jill Su | Software Development Lead (Deputy #2) |
| Richard Tran | Algorithm Specialist |
| Rohan Bhushan | Database Specialist |
| Brittany Trieu | Quality Assurance Lead |
| Pauline Woo | User Interface Specialist |

# Contents

# Login

## 1   Account Creation

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user can create an account by inputting an email and password for their login. |
| **User Goal** | The user's goal is to create an account to use the application. |
| **Desired Outcome** | The user will have their own personal account to use the application. |
| **Dependencies** | None |
| **Requirements** | SR-1 |
| **Pre-conditions** | The user is on the login page |
| **Post-conditions** | The user has an account, and their data is stored in the database. |
| **Trigger** | The user wants a personal account to be able to use the app and view personalized events and join specific groups. |
| **Workflow** | 1. The user shall click the "New User" button<br><br>2. The frontend shall render Signup.js<br><br>3. The user shall fill out the form<br><br>4. The user shall click the "Sign up" button<br><br>5. The frontend shall send a request to the user controller<br><br>6. The backend shall validate the form using validateUserData()<br><br>7. If the form is valid, a user object is created with createUser()<br><br>8. The frontend shall render Login.js |
| **Alternate Workflow** <br> Invalid Email | 3. The user shall input an invalid email (malformed or not UCSD)<br><br>4. The frontend shall display an error |
| **Alternate Workflow** <br> Empty Text Field | 3. The user does not fill out one of the text fields<br><br>4. The frontend shall output an error and prompt the user to fill out all text fields |
| **Alternate Workflow** <br> Invalid Password | 3. The user does not input a password meeting the requirements<br><br>4. The frontend shall output an error, display to the user the password requirements, and prompt the user to enter a valid password |

## 2   Login

| Priority | High |
|---|---|
| Status | Completed |
| Description | The user can login to their account after one has been created. |
| User Goal | The user's goal is to login to their account and access all of their personal features. |
| Desired Outcome | The user will be able to successfully login and use the application. |
| Dependencies | UC-1 |
| Requirements | SR-2 |
| Pre-conditions | The user is on the login page |
| Post-conditions | The user is now logged into their account and can use the application. |
| Trigger | The user wants to use the application with all of their personal features on it. |
| Workflow | 1. The user shall input their email and password 2. The user shall click the "Login" button 3. The frontend shall render Login.js 4. The user shall fill the form 5. The frontend shall send a request to the user controller 6. The backend shall verify the user's information using authUser() 7. The frontend shall render Home.js |
| Alternate Workflow Wrong Email/Pass | 1. The user shall input a wrong email or password 2. The user shall click the "Login" button 3. The frontend shall send a request to the user controller, which will respond with a failure 4. The frontend shall output an error |

# 3 Logout

| Priority | Medium |
|---|---|
| Status | Completed |
| Description | The user can log out of their account after they have logged in and used the application. |
| User Goal | The user's goal is to successfully log out of their account after using the application. |
| Desired Outcome | The user will be logged out of their account and back at the login page. |
| Dependencies | UC-1, UC-2 |
| Requirements | SR-3 |
| Pre-conditions | The user is logged in and is on the home page |
| Post-conditions | The user is now safely logged out of their account. |
| Trigger | The user is done using the application and wants to secure their information by logging out. |
| Workflow | 1. The user shall click on the profile icon on the navbar<br><br>2. The frontend shall render Profile.js<br><br>3. The user shall click on the "Logout" button<br><br>4. The frontend shall send a request to the user controller<br><br>5. The backend shall call signOut()<br><br>6. The frontend shall render Start.js |

# 4 Password Recovery

| Priority | Medium |
|---|---|
| Status | Completed |
| Description | The system will send a password reset email to the user if the user clicks "Forgot Password?" on the login screen of the application. |
| User Goal | The user has forgotten their password to login to their account, and needs their password reset. |
| Desired Outcome | The user can change their forgotten password to a new, known password. |
| Dependencies | UC-1 |
| Requirements | SR-4 |
| Pre-conditions | The user is on the login page |
| Post-conditions | The user has a new, known password after clicking the link sent to their email. |
| Trigger | The user does not remember their password. |
| Workflow | 1. The user shall click the "Forgot Password?" button<br><br>2. The frontend shall render the reset password form<br><br>3. The user shall input their login email<br><br>4. The user shall click the "Send" button<br><br>5. The frontend shall send a request to the user controller<br><br>6. The backend shall call resetPassword() |
| Alternate Workflow<br>Invalid Email | 1. The user shall input an invalid email (does not exist or malformed)<br><br>2. The user shall click the "Send email" button<br><br>3. The frontend shall send a request to the user controller, which will be responded with a failure<br><br>4. The frontend shall display an error and prompt the user to input a valid email |

# Group Management

## 5 Create Group

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user shall be able to create a group. |
| **User Goal** | The user wants to create a group. |
| **Desired Outcome** | The user is in its newly created group. |
| **Dependencies** | UC-1, UC-2 |
| **Requirements** | SR-5 |
| **Pre-conditions** | The user is logged in and is on the home page |
| **Post-conditions** | The user is in a group. |
| **Trigger** | The user wants to create a group with other users. |
| **Workflow** | 1. The user shall navigate to the "My Groups" page (3rd icon on navbar) <br><br> 2. The front end shall display MyGroups.js <br><br> 3. The user shall tap on the create group icon on upper right corner <br><br> 4. The frontend shall render GroupForm.js <br><br> 5. The user shall fill in the form <br><br> 6. The frontend shall send the contents to the group controller <br><br> 7. The backend shall call createGroup() <br><br> 8. The frontend shall render myGroup.js with the new group displayed |
| **Alternate Workflow** <br> Don't create group | 5. The user shall click on the "back" button <br><br> 6. The frontend shall render home.js |

# 6  Join Group

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user shall be able to join a group. |
| **User Goal** | The user wants to join a group. |
| **Desired Outcome** | The user joins a group. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-9 |
| **Requirements** | SR-6 |
| **Pre-conditions** | The user is logged in and is on the home page. |
| **Post-conditions** | The user is in the selected group. |
| **Trigger** | The user wants to join a group. |
| **Workflow** | 1. The user shall use UC-9 to get a list of groups<br><br>2. The user shall click on a specific group<br><br>3. The frontend shall render DummyGroup.js<br><br>4. The user shall click the "Join now" button<br><br>5. The frontend shall send a request to the group controller<br><br>6. The backend shall call addUserToGroup() |
| **Alternate Workflow** | N/A |

# 7 Edit Group

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can edit the details/settings of the group they created. |
| **User Goal** | The user wants to change important details or settings of the group. |
| **Desired Outcome** | The user successfully updates details and settings of the group, and the change is present on the screen of other users. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10 |
| **Requirements** | SR-7 |
| **Pre-conditions** | The user is logged in and is on the home page. |
| **Post-conditions** | The details and settings of the specific group is changed for every user of the app. |
| **Trigger** | The user wants to change the privacy settings of the group. The user doesn't want to present misleading information to other users. |
| **Workflow** | 1. The user shall use UC-10 to view the list of groups<br><br>2. The user shall click on a specific group<br><br>3. The frontend shall render DummyGroup.js<br><br>4. The user shall click on the "Edit" button<br><br>5. The frontend shall render GroupEditForm.js<br><br>6. The user shall fill out the form and click the "Done" button<br><br>7. The frontend shall send the information to the group controller<br><br>8. The backend shall call updateGroup()<br><br>9. The frontend shall rerender DummyGroup.js with the updated information |
| **Alternate Workflow** | N/A |

# 8  Delete Group

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can delete a group. |
| **User Goal** | The user no longer needs the group. |
| **Desired Outcome** | The group is deleted |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-7, UC-10 |
| **Requirements** | SR-8 |
| **Pre-conditions** | The user is logged in and is on the home page. The user has administrative privileges of the designated group |
| **Post-conditions** | The group is deleted and will no longer appear on any user's version of the application. |
| **Trigger** | The group is no longer active. The members do not socialize together anymore. |
| **Workflow** | 1. Use UC-7 to view the edit group form<br><br>2. The user shall click the trash icon<br><br>3. The frontend shall send a request to the group controller<br><br>4. The backend shall call deleteGroup()<br><br>5. The frontend shall render MyGroups.js |
| **Alternate Workflow** | N/A |

# 9    Search Groups

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user can view all groups that have been created publicly. |
| **User Goal** | The user wants to view all the groups that have been created. |
| **Desired Outcome** | The user is presented with a list of groups that may interest them. |
| **Dependencies** | UC-1, UC-2, UC-5 |
| **Requirements** | SR-9 |
| **Pre-conditions** | The user is logged in and is on the home page |
| **Post-conditions** | The user can now browse through a list of publicly created groups. |
| **Trigger** | The user wants to see all the possible groups to meet people who share similar interests. |
| **Workflow** | 1. The user shall tap on the search icon on the navbar<br><br>2. The frontend shall render Search.js<br><br>3. The user shall fill in the search box with the group name or an empty string<br><br>4. The frontend shall send a request to the group controller<br><br>5. The backend shall list all the groups that match the sent in string using getGroups()<br><br>6. The frontend shall display all the given groups |
| **Alternate Workflow** | N/A |

# 10 View Group Page

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user can view all groups that they have joined. |
| **User Goal** | The user wants to view all the groups that they are a member of. |
| **Desired Outcome** | The user is presented with a list of groups that they are a member of. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6 |
| **Requirements** | SR-10 |
| **Pre-conditions** | The user is logged in and is on the home page. |
| **Post-conditions** | The user can now browse through a list of joined groups (if any). |
| **Trigger** | The user wants to see all the groups they are a member of. |
| **Workflow** | 1. The user shall navigate to the "My Groups" page<br><br>2. The frontend shall send a request to the user controller<br><br>3. The backend shall call getUserInfo() and send back the data<br><br>4. The frontend shall send a request to the group controller for each user group<br><br>5. The backend shall call getGroupInfo()<br><br>6. The frontend shall display the list of groups<br><br>7. The user shall tap on the group they want to view<br><br>8. The frontend shall render DummyGroup.js |
| **Alternate Workflow** | N/A |

# 11 View Group Members

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can view the members of a group they are part of. |
| **User Goal** | The user wants to know who is in the same group as them, and/or view their profiles. |
| **Desired Outcome** | The user will be able to view their group's members. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10 |
| **Requirements** | SR-11 |
| **Pre-conditions** | The user is logged in and is on the home page. The user is a member of the designated group |
| **Post-conditions** | The user can see all the members of the group. |
| **Trigger** | The user wants to look up who is part of the group(s) they are in |
| **Workflow** | 1. The user shall use UC-10 to view a group<br><br>2. On the group page is the list of members |
| **Alternate Workflow** | N/A |

# 12   Leave Group

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can leave a group they have joined. |
| **User Goal** | The user no longer wants to be within the group. |
| **Desired Outcome** | The user is no longer a member of the group and does not see group posts. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10 |
| **Requirements** | SR-12 |
| **Pre-conditions** | The user is a member of the designated group |
| **Post-conditions** | The user is no longer listed as a member of that group. The user no longer receives notifications for that group. |
| **Trigger** | The user wants to leave a group because they don't want to receive notifications for that group anymore. |
| **Workflow** | 1. The user shall use UC-10 to view a group<br><br>2. The user shall click the "Leave Group" button<br><br>3. The frontend shall send a request to the group controller<br><br>4. The backend shall call removeUserFromGroup()<br><br>5. The frontend shall render MyGroups.js |
| **Alternate Workflow** | N/A |

# User Management

## 13  Invite User

| | |
|---|---|
| **Priority** | Low |
| **Status** | Not Completed |
| **Description** | A user in a group can invite another user to join the group. |
| **User Goal** | To invite someone else to join the group |
| **Desired Outcome** | Someone else successfully joins the group |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6 |
| **Requirements** | SR-13 |
| **Pre-conditions** | The user is logged in and is on the designated group page. |
| **Post-conditions** | The group contains a new member, the one invited |
| **Trigger** | User wants someone to join the group |
| **Workflow** | 1. The user shall click on the "Invite User" button<br><br>2. The frontend shall render an invitation form<br><br>3. The user shall type the invitee's email and click "Invite" button<br><br>4. The frontend shall send a request to the user controller<br><br>5. The backend shall call inviteUser() |
| **Alternate Workflow**<br>Invalid email | 3. The user shall type in an email that does not exist<br><br>4. The frontend shall make the invite button unclickable and request that a valid email is entered |

# 14 Remove User

| | |
|---|---|
| **Priority** | Low |
| **Status** | Not Completed |
| **Description** | A user in a group can remove another user in the same group. |
| **User Goal** | To remove someone from the group. |
| **Desired Outcome** | The removed person will no longer be in the group. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10, UC-11 |
| **Requirements** | SR-14 |
| **Pre-conditions** | The user is logged in and is on the designated group page. The user has administrative privileges |
| **Post-conditions** | The group contains one less member, the one removed. |
| **Trigger** | User wants to remove a member. |
| **Workflow** | 1. The user shall use UC-11 to view the group members<br><br>2. The user shall tap the 'X' button next to the desired member to be removed<br><br>3. The frontend shall send a request to the group controller<br><br>4. The backend shall call removeUser()<br><br>5. The frontend shall rerender DummyGroup.js |
| **Alternate Workflow** | N/A |

## 15   View User Profile

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Not Completed |
| **Description** | A user is allowed to view the profile of another user. |
| **User Goal** | To view someone else's profile. |
| **Desired Outcome** | The system shows the profile of the searched user. |
| **Dependencies** | UC-1, UC-2, UC-6, UC-11, UC-33 |
| **Requirements** | SR-15 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | The profile of the desired user is shown. |
| **Trigger** | A user wants to look at someone else's profile |
| **Workflow** | 1. The user shall use UC-11 to view the group members<br><br>2. The user shall click on a member<br><br>3. The frontend shall present the user a list of actions<br><br>4. The user shall click "View profile"<br><br>5. The frontend shall send a request to the user controller<br><br>6. The backend shall call getUserInfo()<br><br>7. The frontend shall render DummyUser.js |
| **Alternate Workflow** | N/A |

# 16 Create Post

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | A user is allowed to post inside a group. |
| **User Goal** | To publicize something within a group. |
| **Desired Outcome** | The system shall display the user's post for all group members to see. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10 |
| **Requirements** | SR-16 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | The post is shown on the group page and is updated on all other users' accounts. |
| **Trigger** | A user has some important announcement relevant to the group. |
| **Workflow** | 1. The user shall use UC-10 to view a group<br><br>2. The user shall click on "Add Post (+)"<br><br>3. The frontend shall render PostForm.js<br><br>4. The user shall fill out the form and click on the "Create Post" button<br><br>5. The frontend shall send a request the group controller<br><br>6. The backend shall call createTask()<br><br>7. The frontend shall render DummyGroup.js |
| **Alternate Workflow** | N/A |

## 17 Edit Post

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will edit a post that they created. |
| **User Goal** | The user wants to change the title or body of a post they previously created. |
| **Desired Outcome** | The details of a post the user created will be changed. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-16 |
| **Requirements** | SR-17 |
| **Pre-conditions** | The user is logged in and on a group page. One of the members in the group has previously created a post. |
| **Post-conditions** | The post details are changed. |
| **Trigger** | A post the user has created needs to be updated with new information. |
| **Workflow** | 1. The user shall tap on one of the posts under the "Posts" section of the group page<br><br>2. The frontend shall render PostPopUp.js<br><br>3. The user shall click on the "Edit" button<br><br>4. The frontend shall render the PostEditForm.js<br><br>5. The user shall update the post information as necessary<br><br>6. The frontend shall send a request to the group controller<br><br>7. The backend shall call editPost()<br><br>8. The frontend shall render DummyGroup.js |
| **Alternate Workflow**<br>Do not edit post | 5. The user shall tap on the back button<br><br>6. The frontend shall navigate back to the group page |

# 18 Delete Post

| Priority | Medium |
|---|---|
| Status | Completed |
| Description | The user can delete a post. |
| User Goal | The user no longer wants the post. |
| Desired Outcome | The post is deleted. |
| Dependencies | UC-1, UC-2, UC-5, UC-6, UC-16, UC-17 |
| Requirements | SR-18 |
| Pre-conditions | The user is logged in. The user is a member of the group. The user is on the group page |
| Post-conditions | The post is deleted and will no longer appear on any user's version of the application. |
| Trigger | The post has misleading information or is no longer relevant. |
| Workflow | 1. The user shall use UC-17 to go to edit page of a post<br><br>2. The user shall tap on the trash icon<br><br>3. The frontend shall send a request to the group controller<br><br>4. The backend shall call deletePost()<br><br>5. The frontend shall render DummyGroup.js |
| Alternate Workflow | N/A |

# 19 View Post

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will be able to view the post along with its details. |
| **User Goal** | The user wants to see details of the post. |
| **Desired Outcome** | The user can see their post and its details. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-16 |
| **Requirements** | SR-19 |
| **Pre-conditions** | The user is logged into their account, in a group. Posts have been made in that group. The user is on the group page |
| **Post-conditions** | The system will display the details of the posts that have been made. |
| **Trigger** | The user wants to view their posts and their details. |
| **Workflow** | 1. The user shall tap on a specific post<br><br>2. The frontend shall render PostPopUp.js |
| **Alternate Workflow** <br> Stop viewing | 3. The user shall tap on the back button<br><br>4. The frontend shall navigate back to the group page |

# 20 Make a Group Member an Admin

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Not Completed |
| **Description** | A user who has created a group is allowed to promote a member to admin status. |
| **User Goal** | The user wants another admin to help manage the group. |
| **Desired Outcome** | The group owner has granted another member administrative privileges. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-11, UC-15 |
| **Requirements** | SR-20 |
| **Pre-conditions** | The user is logged in and is on the designated group page. Moreover, the user is the owner of the group |
| **Post-conditions** | The chosen member will now be able to perform administrative functions. |
| **Trigger** | The user needs help to help manage the group. |
| **Workflow** | 1. The user shall use UC-15 to view the profile of a group member<br><br>2. The user shall click "Make user admin"<br><br>3. The frontend shall send a request to the group controller<br><br>4. The backend shall call makeUserAdmin() |
| **Alternate Workflow** | N/A |

# Interaction Management

## 21   Create Event

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user creates a new event happening on campus. |
| **User Goal** | The user wants to create a new event. |
| **Desired Outcome** | A new event is created. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6 |
| **Requirements** | SR-18 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | The event is created and added to a list of group events. |
| **Trigger** | A new event has been organized by the school or a club/organization and they want to increase attendance. |
| **Workflow** | 1. The user shall click "Add Event (+)" button<br><br>2. The frontend shall render EventForm.js<br><br>3. The user shall fill out the form<br><br>4. The frontend shall send a request to the group controller<br><br>5. The backend shall call createEvent()<br><br>6. The frontend shall render DummyGroup.js |
| **Alternate Workflow** Incomplete fields | 5. The frontend shall display an error that some fields are blank |
| **Alternate Workflow** Don't create event | 3. The user shall click the "Back" button<br><br>4. The frontend shall renavigate back to the group page |

## 22    Edit Event

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will edit an event that they created. |
| **User Goal** | The user wants to change the time, date, location, or description of an event they previously created. |
| **Desired Outcome** | The details of an event the user created will be changed. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-21 |
| **Requirements** | SR-22 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | The event details are changed and updated in the database. |
| **Trigger** | An event the user has created has changed time, date, location, or has been cancelled. |
| **Workflow** | 1. The user shall click on a specific event<br><br>2. The frontend shall render EventPopUp.js<br><br>3. The user shall click the "Edit" button<br><br>4. The frontend shall render EventEditForm.js<br><br>5. The user shall enter the updated information<br><br>6. The user shall click the "Done" button<br><br>7. The frontend shall send a request to the group controller<br><br>8. The backend shall call editEvent()<br><br>9. The frontend shall render DummyGroup.js |
| **Alternate Workflow**<br>Don't edit | 5. The user shall click the back button<br><br>6. The frontend shall renavigate to the group page |

## 23   Delete Event

| Priority | Medium |
|---|---|
| **Status** | Completed |
| **Description** | User deletes an existing event. |
| **User Goal** | User deletes an existing event from the list of group events. |
| **Desired Outcome** | Event will no longer exist. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-21, UC-22 |
| **Requirements** | SR-23 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | The event is no longer visible and is removed from the database. |
| **Trigger** | The user's event is cancelled or has passed. |
| **Workflow** | 1. The user shall use UC-22 to view the event edit form <br><br> 2. The user shall click the trash icon <br><br> 3. The frontend shall send a request to the group controller <br><br> 4. The backend shall call deleteEvent() <br><br> 5. The frontend shall render DummyGroup.js |
| **Alternate Workflow** <br> Don't delete | 2. The user shall click the back button <br><br> 3. The frontend shall renavigate to the group page |

## 24 Share Event

| | |
|---|---|
| **Priority** | Low |
| **Status** | Not Completed |
| **Description** | The user can share an event with others. |
| **User Goal** | User wants to notify others about a future event. |
| **Desired Outcome** | User sends a notification to selected group members. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-21 |
| **Requirements** | SR-21 |
| **Pre-conditions** | The user is logged in and is on the designated group page |
| **Post-conditions** | Selected event details are sent to other users that the user chooses. |
| **Trigger** | The user wants to let others know about a new or interesting event. |
| **Workflow** | 1. The user shall use UC-25 to view an event<br><br>2. The user shall click the "Share" button<br><br>3. The frontend shall render a drop down menu of members<br><br>4. The user shall check off the members to share the event to<br><br>5. The user shall click the "Done" button<br><br>6. The frontend shall send a request to the user controller<br><br>7. The backend shall call shareEventToUser() |
| **Alternate Workflow** Don't share | 4. The user clicks on the back button<br><br>5. The frontend shall close the sharing screen |
| **Alternate Workflow** No user selected | 6. If the user did not select anyone to share, the frontend shall prompt the user to select at least one person |

## 25   View Event

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will be able to view the event along with its details. |
| **User Goal** | The user wants to see details of the event. |
| **Desired Outcome** | The user can see their event and its details. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-21 |
| **Requirements** | SR-25 |
| **Pre-conditions** | The user is logged into their account. Events have been made and assigned. The user is on the home page. |
| **Post-conditions** | The system will display the details of the events that have been made. |
| **Trigger** | The user wants to view the event and the details. |
| **Workflow** | 1. The user shall click on an event<br><br>2. The frontend shall render EventPopUp.js |
| **Alternate Workflow** Don't share | 3. The user clicks on the "X" icon<br><br>4. The frontend shall close the popup |
| **Alternate Workflow** View group event | 1. The user shall use UC-10 to view a group<br><br>2. The user shall click on an event<br><br>3. The frontend shall render EventPopUp.js |

# 26 Send Notification/Reminders

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Not Completed |
| **Description** | The user can send a notification for an event or reminder for a task |
| **User Goal** | The user wants group members to receive a push notification from the app. |
| **Desired Outcome** | The other user receives the notification and attends the event or completes the task. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-21 |
| **Requirements** | SR-26 |
| **Pre-conditions** | The user is logged in and is on the designated group page. Moreover, the user is the group owner. |
| **Post-conditions** | A notification pops up for all group members with notifications on. |
| **Trigger** | The user wants to make sure the other user remembers an upcoming event or task. |
| **Workflow** | 1. The user shall click on "Send notification" button<br><br>2. The frontend shall render a schedule event form<br><br>3. The user shall input the necessary information (message, date, scheduling system)<br><br>4. The user shall click "Done"<br><br>5. The frontend shall display a confirmation message<br><br>6. The user shall click "yes" or "no" accordingly<br><br>7. The frontend shall send the contents to the group controller<br><br>8. The backend shall send the notification to the group members if "yes" is clicked by calling sendNotificaiton() |
| **Alternate Workflow** Schedule Notification | 5. The user shall input a time and date<br><br>6. The frontend shall display a confirmation message<br><br>7. The user shall click "yes" or "no" accordingly<br><br>8. The frontend shall send the contents to the group controller<br><br>9. The backend shall schedule the notification is "yes" is clicked by calling sendReminder() |

## 27   Commenting on Post

| | |
|---|---|
| **Priority** | Low |
| **Status** | Not Completed |
| **Description** | The user can make a comment on an event that has been posted |
| **User Goal** | The user wants to make a comment about a particular post |
| **Desired Outcome** | The user's message is sent and other users can read it |
| **Dependencies** | UC-1, UC-2, UC-6, UC-10, UC-16 |
| **Requirements** | SR-27 |
| **Pre-conditions** | The user is in a group with at least one post |
| **Post-conditions** | The user successfully sends a comment and it is displayed on every other user's screen |
| **Trigger** | The user wants to communicate with the rest of the group about a specific event |
| **Workflow** | 1. The user shall use UC-10 to view a specific group<br><br>2. The user shall click on an event<br><br>3. The frontend shall render EventPopUp.js<br><br>4. The user shall click the "Comment" button<br><br>5. The frontend shall render CommentForm.js<br><br>6. The user shall fill out the form and click "Done"<br><br>7. The frontend shall send the contents to the group controller<br><br>8. The backend shall call EditEvent()<br><br>9. The frontend shall rerender EventPopUp.js |
| **Alternate Workflow** | N/A |

# Task Management

## 28   Create Task

| | |
|---|---|
| **Priority** | High |
| **Status** | Completed |
| **Description** | The user can create tasks within a group. |
| **User Goal** | The user wants to organize and assign tasks that must be completed within a group. |
| **Desired Outcome** | The user who assigns a task and the rest of the group members will be able to see the tasks. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10 |
| **Requirements** | SR-24 |
| **Pre-conditions** | The user is logged in and in at least one group. The user is on the home page ( or any page with navigation bar). |
| **Post-conditions** | A new task is assigned and displayed on the tasks page. |
| **Trigger** | The user wants to assign a task for a group member/members to do. |
| **Workflow** | 1. The user shall use UC-10 to view a group<br><br>2. The user shall click "Add Task (+)"<br><br>3. The frontend shall render TaskForm.js<br><br>4. The user shall fill in the form and click the "Done" button<br><br>5. The frontend shall send a request to the task controller<br><br>6. The backend shall call createTask()<br><br>7. The frontend shall render DummyGroup.js |
| **Alternate Workflow**<br>Cancel task | 4. The user shall tap on the back button<br><br>5. The frontend shall renavigate to the group page |
| **Alternate Workflow**<br>Empty required fields | 6. The frontend shall display error message that some required fields are empty |

# 29 Edit Task

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can edit a task. |
| **User Goal** | The user wants to make changes to the task's name, description, expected deadline, and/or group member(s). |
| **Desired Outcome** | The changes made to the task will be reflected on the tasks page for all members of the group to see. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10, UC-28 |
| **Requirements** | SR-29 |
| **Pre-conditions** | The user is logged in and in at least one group. The user is on the home page ( or any page with navigation bar). |
| **Post-conditions** | The task will be edited and updated on the tasks page. |
| **Trigger** | There is a mistake on the task that the user wishes to change, or the information initially filled in is not relevant anymore and the user desires to edit the tasks to reflect the updated information. |
| **Workflow** | 1. The user shall use UC-10 to view a group page |
| | 2. The user shall click on a specific task |
| | 3. The frontend shall display TaskPopUp.js |
| | 4. The user shall click the "Edit" button |
| | 5. The frontend shall display TaskEditForm.js |
| | 6. The user shall update the task information and click the "Done" button |
| | 7. The frontend shall send a request toe the task controller |
| | 8. The backend shall call editTask() |
| | 9. The frontend shall render DummyGroup.js |
| **Alternate Workflow** Don't edit | 6. The user shall click the back button |
| | 7. The frontend shall renavigate to the group page |

# 30   Delete Task

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can delete a task. |
| **User Goal** | The user has completed the task or no longer needs to complete the task. |
| **Desired Outcome** | The task has been deleted. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10, UC-28, UC-29 |
| **Requirements** | SR-30 |
| **Pre-conditions** | The user is logged in and in at least one group. The user is on the home page ( or any any page with navigation bar) |
| **Post-conditions** | The task will no longer be displayed in the tasks page. |
| **Trigger** | The user no longer needs to complete the task and doesn't want to see it anymore on the group page. |
| **Workflow** | 1. The user shall use UC-10 to view a group page<br><br>2. The user shall use UC-29 to view the edit form<br><br>3. The user shall click the trash icon<br><br>4. The frontend shall send a request to the task controller<br><br>5. The backend shall call deleteTask()<br><br>6. The frontend shall render DummyGroup.js |
| **Alternate Workflow** <br> Don't delete | 3. The user shall click the back button<br><br>4. The frontend shall renavigate back to the group page |

# 31 View Tasks

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will be able to view the tasks assigned along with its details. |
| **User Goal** | The user wants to see details of the tasks related to them. |
| **Desired Outcome** | The user can see their task(s) and its details. |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10, UC-28 |
| **Requirements** | SR-31 |
| **Pre-conditions** | The user is logged into their account, in a group with tasks that have been made and assigned. The user is on the home page. |
| **Post-conditions** | The system will display the details of the tasks that have been assigned. |
| **Trigger** | The user wants to view their tasks and its details. |
| **Workflow** | 1. The user shall click on a specific task<br><br>2. The frontend shall render TaskPopUp.js |
| **Alternate Workflow** Stop viewing details | 3. The user shall click on the "X" button<br><br>4. The frontend shall close the task pop up |
| **Alternate Workflow** View group task | 1. The user shall use UC-10 to view a group<br><br>2. The user shall click on a specific task<br><br>3. The frontend shall render TaskPopUp.js |

# 32   Complete Task

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user will be able to complete the task and remove it from the list. |
| **User Goal** | The user wants to mark the task as complete. |
| **Desired Outcome** | The user wants to remove the task from the list because it is completed |
| **Dependencies** | UC-1, UC-2, UC-5, UC-6, UC-10, UC-28, UC-31 |
| **Requirements** | SR-32 |
| **Pre-conditions** | The user is logged into their account and in a group. Tasks have been made and assigned. The user is on the home page (or any page with navigation bar). |
| **Post-conditions** | The system will no longer display the task because it is completed |
| **Trigger** | The user has completed the task and no longer needs to see it. |
| **Workflow** | 1. The user shall use UC-10 to view a group<br><br>2. The user shall use UC-31 to view a task<br><br>3. The user shall tap the "Complete" button<br><br>4. The frontend shall not display the task |
| **Alternate Workflow** Do not complete | 3. The user shall click on the "X" button<br><br>4. The frontend shall close the task pop up |

# Profile Management

## 33    View Profile

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can view their personal profile. |
| **User Goal** | The user wants to view their own personal profile information. |
| **Desired Outcome** | The user's personal information is displayed. |
| **Dependencies** | UC-1, UC-2 |
| **Requirements** | SR-33 |
| **Pre-conditions** | The user has an account and is logged in. The user is on the home page (or any page with navigation bar) |
| **Post-conditions** | The app changes accordingly to the user's personal information. |
| **Trigger** | The user has preferences on what their personal bio shows and how their app should run. |
| **Workflow** | 1. The user shall click on the profile icon on the navbar (4th icon) 2. The frontend shall render Profile.js 3. The frontend shall send a request to the user controller 4. The backend shall call getUserInfo() 5. The frontend shall display the user information |
| **Alternate Workflow** | N/A |

# 34   Edit Profile

| | |
|---|---|
| **Priority** | Medium |
| **Status** | Completed |
| **Description** | The user can edit personal settings. |
| **User Goal** | The user wants to change specific settings such as notifications or personal bio. |
| **Desired Outcome** | The settings such as personal bio or notifications are successfully updated. |
| **Dependencies** | UC-1, UC-2 |
| **Requirements** | SR-34 |
| **Pre-conditions** | The user has an account and is logged in. The user is on the home page (or any page with navigation bar). |
| **Post-conditions** | The app changes accordingly to the preferences of the user. |
| **Trigger** | The user has preferences on what their personal bio shows and how their app should run. |
| **Workflow** | 1. The user shall click on the profile icon on the navbar (4th icon) <br><br> 2. The frontend shall render Profile.js <br><br> 3. The user shall click on the pencil icon <br><br> 4. The frontend shall render ProfileForm.js <br><br> 5. The user shall update information as necessary <br><br> 6. The user shall click on the "Done" button <br><br> 7. The frontend shall send a request to the user controller <br><br> 8. The backend shall call updateUser() <br><br> 9. The frontend shall rerender Profile.js |
| **Alternate Workflow** <br> Don't update | 5. The user clicks the back button <br><br> 6. The frontend renavigates to the profile page |

# 35 Delete Account

| | |
|---|---|
| **Priority** | Low |
| **Status** | Completed |
| **Description** | The user will be able to delete their account |
| **User Goal** | The user no longer needs an account since they no longer use the app |
| **Desired Outcome** | The user has successfully deleted their account |
| **Dependencies** | UC-1, UC-2, UC-34 |
| **Requirements** | SR-29 |
| **Pre-conditions** | The user has the app downloaded and has an account created. The user is logged into their account. The user is on the home page (or any page with navigation bar). |
| **Post-conditions** | The user has their account removed from the system. |
| **Trigger** | The user no longer goes to UC San Diego or finds that they no longer have a use for the app. |
| **Workflow** | 1. The user shall use UC-34 to view the edit profile form<br><br>2. The user shall click the "Delete Account" button<br><br>3. The frontend shall send a request to the user controller<br><br>4. The backend shall call deleteUser()<br><br>5. The frontend shall render Start.js |
| **Alternate Workflow**<br>Do not delete | 2. The user shall click the back button<br><br>3. The frontend shall renavigate to the profile page |