Use Case Name: User clicks button to access settings page

Preconditions: User already logged in, and User has already opened the menu containing the settings button

Main Success Scenario:

1. User clicks settings button
2. User is brought to page containing list of settings

Extensions: N/A

Miscellaneous: N/A

Use Case Name: User clicks "favorite" button for a club

Preconditions: User is already logged in, and User is viewing a list of clubs page, User is on the page of a club, or User is viewing a list of their favorited clubs

Main Success Scenario:

The club is now added to the user's list of favorite clubs

1. User is viewing club profile
2. User clicks "favorite" button
3. User has now added club to its favorited list

Extensions:

    3a. User has removed the club from favorited list

Miscellaneous: The button acts as a togglable

Use Case Name: User views favorited clubs

Preconditions: User is already logged in

Main Success Scenario:

1. User clicks link in profile to view list of favorited clubs

2. User is brought to page listing favorited events

Extensions:

2a. User is brought to empty page explaining they have no favorited clubs

Miscellaneous:

Use Case Name: User clicks link to view list of clubs

Preconditions: User is already logged in

Main Success Scenario:

1. User clicks link to view clubs
2. User brought to page with all clubs

Extensions: N/A

Miscellaneous: N/A

Use Case Name: User clicks link to view club profile

Preconditions: User is viewing a list of clubs page, or User is viewing a list of their favorited clubs

Main Success Scenario:

1. User clicks link to view club profile
2. User is brought to club profile page

Extensions:

2a. User is brought to page explaining that the club does not exist/has been removed

Miscellaneous: N/A

Use Case Name: User clicks button to view event photo gallery

Preconditions: User's GPS location is at the event or was recently at the event

Main Success Scenario:

1. User clicks button to view event gallery
2. User brought to event gallery page

Extensions:

2a. User brought to page explaining event has not/already occurred

2b. User is brought to page where there are no photos yet

Miscellaneous: N/A

Use Case Name: User selects to silence specific notifications

Preconditions: User has notifications enabled, and is logged in

Main Success Scenario:

1. User selects to view a particular club profile
2. On the profile, the user selects the option to silence this particular club's notifications.
3. The user will no longer receive notifications about this club's events

Extensions: N/A

Miscellaneous: Note that user will have the option to turn them back on (the workflow for this is identical and obvious)

Use Case Name: User selects recover password/username

Preconditions: User must have had a preexisting account registered under a valid email

Main Success Scenario:

1. User selects "Recover Account"
2. They are prompted for their email address.
3. The user enters their email address.
4. System alerts the user that they should receive and email with instructions (TBA) to reset their password or username
5. User follows such instructions in their inbox to do so
6. User is prompted to log in with new credentials

Extensions:

    4a. User did not receive the email

        1. System allows user to re-send the email

Miscellaneous: N/A

Use Case Name: User creates an account

Preconditions: User has access to a valid email address

Main Success Scenario:

1. User select "Register"
2. User is prompted to enter in their full name, email address, desired username, and desired password. Prompt for re-entry (e.g. for password) as necessary.
3. User enters their details and submits the form.
4. An account with the submitted credentials is registered in the database.
5. User now has an account

Extensions:

Various potential extensions here (see Miscellaneous)

    2a. Password and re-entered password do not match

        1. Prompt for correction

    2b. Password does not meet strength requirements

        1. Prompt for correction

    2c. Email address is already registered

        1. Prompt for login

Miscellaneous:

1. After a user creates an account, it might be convenient that they are automatically logged in.
2. Password requirements? Special characters, numbers, length, etc.

Use Case Name: User edits account/profile details

Preconditions: User has an account, is logged in

Main Success Scenario: NOTE: Profile/Account includes all of the details related to the user - including their login credentials (username, email, password) as well as information not related to login (full name, interests)

1. User selects "View Account" or "View Profile". This is viewable only by them.
2. User is brought to a page displaying various information about themself
3. User is able to update any information on this page
4. User selects to save any changes
5. Internal representation of the user and their information is updated

Extensions: N/A

Miscellaneous: Updating email address may warrant validation of new email address. Not truly necessary, but is something we could implement if there is time

Use Case Name: User logs into account

Preconditions: User has created an existing account

Main Success Scenario:

1. User clicks on field to type in their username
2. User clicks on next field to type in their password
3. User presses the Login button which checks to validate credentials
4. User is successfully logged into the main home screen

Extensions:

3a. If the users entered credentials are incorrect the system will display a red error message that reads "Incorrect username or password"

Miscellaneous: N/A

Use Case Name: Admin promotes club member to club leader

Preconditions: There is an existing club page. The user is already a member of that club page.

Main Success Scenario:

1. Admin promotes a user to be the leader of a club
2. The user is given leadership privileges on the club page.

Extensions:

2a. There is already an existing leader of the club that will need to have the privileges removed from the club page before a new leader can be given.

1. Admin must verify the new leader
2. Admin must remove the previous leader

4a. The leader request is determined by the Admin to be invalid

1. Admin will deny the user access to leadership privileges along with possible reason for why the denial was given

Miscellaneous:

Use Case Name: Admin creates club profile (page)

Preconditions: The club has been accepted formally on campus

Main Success Scenario:

1. Admin selects create new club
2. Admin fills out the fields that contain the non editable information of a club profile
3. Admin makes the club profile publicly viewable

Extensions: N/A

Miscellaneous: N/A

Use Case Name: Club leader can edit club profile

Preconditions: User has been promoted to club leader.

Main Success Scenario:

1. User (club leader) navigates to club profile.
2. User is shown the profile in a view that is public.
3. User has option to click an icon such as a pencil that will allow the fields on the page to become editable.
4. The user can change the fields that are open for editing.

Extensions: N/A

Miscellaneous: Not all fields should be editable, for example the name of the Club that has been approved by the school needs to remain unless formally changed with the school first.

(Robert)

Use Case Name: User selects recommended clubs

Preconditions: User has filled out an interest form, is logged in.

Main Success Scenario:

1. User clicks "Recommended clubs".
2. System displays a list of clubs.
3. User clicks on a club that interests them.
4. System displays a detailed description of the club, upcoming events, locations, and times.
5. User clicks on an event for that club.
6. System goes to map view, showing the event location on the map overlay.

Extensions:

*a. If at any time, the user clicks on the X button on the upper right.

1. The system goes back to the main map view.

2a. If there are no recommended clubs, the system displays a background of the list view saying "Nothing here yet! Set profile interests to populate this list."

Miscellaneous: N/A

(Gaurav)

Use Case Name: User fills in new event form

Preconditions: The user has all relevant info for the event and is able to fill out the form.

User has the link to the form

Note: the user doesn't currently need an account for this.

Main Success Scenario:

1. User clicks link to new event form.
2. User fills out new event form including their email and presses "submit"
3. The system receives the new event form and stores it in the events pending approval table
4. The system checks event against existing events to establish if event already exists
5. The system redirects the user to page stating event has been submitted and is pending approval.
6. The user is sent an email stating that the event was created using their email and a link is included that loads a page with the status of approval for the event. Included in the email is a copy of the form.

Extensions:

4a. The event already exists

1. System sends email to user that the event already exists.
2. System removes the event from the pending approval table

2a. The event is rejected by form filter (stops obviously bad events)

1. System sends user to page saying "unable to create event"

2b. The event form is not not stored.

1. The system sends email to the user stating the event was not created please try again. The email will contain a copy of the form that was submitted.

Miscellaneous:

This implies we need a black list of words that if they are found in the new event form, will trigger an automatic rejection.

(Gaurav)

Use Case Name: Club leader manually creates a new event

Preconditions: The club leader already has signed into their account, belongs to that club, and is the club president/leader who can speak for the club.

Main Success Scenario:

1. Club leader selects the menu bar
2. Club leader selects the club events link
3. The system redirects the user to their club events page
4. The club leader selects the plus button in bottom right of the page
5. The system redirects the club leader to "New Event" form with pre filled in Club section
6. The club leader fills in the "New Event" form and presses submit.
7. The system calls the update events table command to update the database
8. The system displays timed message; "New event: [event name] has been created"
9. The timed message ends and the club events page is refreshed by the system with the newly created event showing.

Extensions:

a*. The club leader's app has disconnected from the server

1. The system displays error msg; "You have lost connection.\nPlease try again later"
2. The system redirects the user to the home page.

7a. System is unable to update the events database

1. The system saves the form locally in drafted events and pushses the update command onto the "pedning" requests stack
2. The system displays error msg; "ERROR: event not created, event saved in drafts"

Miscellaneous:

This requires a stack for pending requests, saves info to be sent again.

This means there is local storage being used.

There drafts section for new events.

(Gaurav)

Use Case Name: User RSVPs to an event

Preconditions: User is signed into their account, and has navigated to the event page.

Main Success Scenario:

1. The user selects "I will be attending" button name tentative.
2. The system sends update to the database to increase the number of attendees
3. The system refreshes the event page, with the "I will be attending" button now saying "see you soon"

Extensions:

2a. The system is unable to update the database to reflect the new number of attendees

1. The system will resend the attempt to update the database.

2b. The system is unable to update the database to reflect the new number of attendees, 2nd attempt

1. The system local pushes the request command onto a stack of "pending" requests.
2. The system displays error msg to the user; "Cannot RSVP at this time"

2c. The event is full

1. The system will return a msg; "The event is fully booked"

2d. The event has expired

1. The system displays error msg: " [event name] took place at [event time] and has concluded"

Miscellaneous:

This requires a stack for pending requests, saves info to be sent again.

(Gaurav)

Use Case Name: User clicks "Map"

Preconditions: User is already signed in.

Main Success Scenario:

1. User clicks the side bar and opens the menu.
2. System displays the menu, with the a link to the map
3. User selects map option.
4. System displays the map.

Extensions:

4a. The system is unable to display the map.

> 1. The system displays an error page

Miscellaneous:

(Robert)

Use Case Name: User wants to share a photo of an event

Preconditions: User is on event detail screen

Main Success Scenario:

1. User clicks "Share photo."
2. System displays icons for source of photo
3. User clicks my photos app
4. System send system call to phone to open photos app
5. User selects a photo, or many photos.
6. System uploads photos to the event photo directory.
7. System displays a photo directory with uploaded photos included.

Extensions:

*a. If at any time the user clicks "cancel".

1. The system returns to the map overlay main screen.

3a. The user clicks my files icon.

1. The system sends a system call to the phone to open the file explorer app.
2. The user selects one or many photo files.
3. The user selects upload.
4. The system uploads the photo files to the event directory.
5. The transaction continues.

Miscellaneous: N/A

(Robert)

Use Case Name: User walks near ongoing/recent/upcoming event

Preconditions: Location access granted. User is within X meters of an event.

Main Success Scenario:

1. User walks into the distance threshold of a near event.
2. The system sends a push notification to the phone.
3. The user taps the notification.
4. The system opens the app, and opens the event detail screen.

Extensions:

2a. The user does not tap the notification.

1. The system adds the event to a list of nearby occurring events, which can be accessed via the nearby events button.
2. System continues to run in background

Miscellaneous: N/A

(Robert)

Use Case Name: User enters Schedule

Preconditions: The user is logged in and has tapped on the Schedule menu button.

Main Success Scenario:

1. The user taps the Modify Schedule button.
2. The system displays a screen showing the current list of scheduled user locations.
3. The user taps add scheduled location.

4. The system displays a form with scheduled location name, start time, end time, and recurrence interval, and delete schedule item button.
5. The user fills out the form and clicks submit.
6. The system adds the scheduled location to the list containing the user's schedule.
7. The system displays the schedule screen.

Extensions:

*a. If at any time the user clicks cancel, the system returns to the main map overlay screen.

3a. The user taps on a scheduled location from the list.

1. The system opens the filled out form for that scheduled location.
2. The user modifies the form as desired.
3. The system updates the scheduled location.

4a. If the user taps delete schedule item button

1. The system displays a screen asking the user to confirm or cancel deletion.
   a. If the user clicks the cancel button.
      i. The system returns to the list of scheduled locations.
   b. The user clicks confirm deletion.
      i. The system deletes the scheduled location and returns to the list of scheduled locations screen.

5a. If the user clicks the X button on the upper right of the form.

1. The system returns to the list of scheduled locations.

Miscellaneous: N/A

Class Structure:

Event

       Attributes: EventID, Location, Time, Description, Photo Gallery, RSVP list

       Methods: getLocation(eventID), getLocation(eventID), getDescription(eventID), getEvent(eventID), editEvent(eventID), deleteEvent(eventID), createEvent(eventID)

Club

       Attributes: clubID, Description,

       Methods: getAllEvents(clubID). getDescription(clubID), getClubID()

Account abstract class

       User Account extends account

       Club Leader extends user account

       Administrator extends account