Usage Model: Scenarios and Use Cases

# **Part 1: Scenarios**

**Scenario 1 - Joining group and viewing usage metrics**

**By:** Michael Nguyen

**Source:** Field notes 44, 4, 45, 6, 49, 50.

Goal models “Improve students’ social connectedness” and “Enhance students’ interests.”

**Assumptions:** I assumed that UCI administration will be able to contact/connect with student group admins.

**Post-Questions:** What is the flow if a student declines a friend request? How and where will students view all of the friend requests they receive?

Dominic has been feeling lonely in his second year of college because he didn’t get the chance to make any friends when his first year was completely online. He hears about Anteater Groups from a UCI email and decides to register with his UCINetID, and edit his profile selecting all of his interests and school related activities. Dominic really loves rock-climbing, so he browses and searches for a rock-climbing group and finds one that he decides to join. Excited to see that a lot of people from UCI enjoy rock-climbing, he sends a friend request to 4 people that are also in their second year. Dominic meets up with his 4 friends and picks a date for them to all go rock-climbing together! He does not feel lonely anymore and feels that he belongs to a community now.

Meanwhile, Benjamin, a UCI Admin, wants to monitor the status of the groups in Anteater Groups. He logs into the application and visits the dashboard to view all group data metrics. Benjamin notices that the rock-climbing group is highly active and has several events planned with a high number of RSVP’s. On the other hand, he reads that crocheting is hardly active and does not have any events planned at all. He reaches out to the student admin of that group and decides to plan out two events with good incentives to get more students to participate in the next month. After three months, Benjamin logs back in and visits the crocheting group to see the amount of engagement. He is super happy when he notices that students in that group are participating more than before and that there are monthly events planned for the remainder of the school year.

**Scenario 2 - Blocking and Removing Students from Groups**

**By:** Jaime Park

**Source:** Field notes 4, 12, 25, 29, 33, 37, 52, 69, Goal model “Maximize individuals' safety and security”

**Assumptions:**

On a student’s profile, there are buttons to befriend, block, and report them.

Students can only privately message each other once they are “friends.”

**Post-Questions:**

How much are administrators allowed to see the reported students’ conversations?

Can administrators directly message students for more information about reports?

Can reported/ban students appeal to overturn the outcome?

Andrea is a second-year UCI student who is starting to get into crocheting. She wants to receive advice on how to improve her crochet projects and to also see what others create, and joins a crocheting group on AnteaterGroups. Andrea introduces herself to the group, and she receives a friend request from a person named Sam. Excited to make her first friend through the group, she accepts the friend request and messages Sam privately. Andrea puts her phone down, and receives a notification that Sam has replied back to her. Sam replies to her greeting, and says that Andrea’s projects are extremely ugly and needs much more work. Sam also brags that they already have a small online crochet business which went viral on social media for its cute items, and that they could teach Andrea how to improve if she promoted the business on her personal social media accounts. Andrea says no, to which Sam starts calling Andrea hateful names and racially-targeted slurs. Andrea becomes upset and disgusted, and immediately clicks the “block” button on Sam’s profile to block them. Sam is now no longer able to privately message Andrea, as they are no longer her friend. Still on Sam’s profile, Andrea clicks the “report” button next to their name to report them, and a new window pops up for Andrea to complete the report. Required with details about the incident, Andrea types up a short statement about the report.

Tracy, a UCI administrator, becomes notified about the report. She opens up an investigation about the report, and views Andrea’s message. She then opens up a preview of Andrea and Sam’s conversation, to which she can see the messages pertaining to the incident. As Tracy has evidence of Sam going against the guidelines of student conduct, she permanently bans Sam from the crocheting group.

**Scenario 3 - Creating and going to an event**

**By:** Gail Manlapaz

**Source:** Field notes 6, 52, 74, 75, 77, 81, and Goal models “Improve students’ social connectedness” and “Improve student involvement with campus resources and organizations”

**Assumptions:**

To RSVP for an event, a student can simply click on the event notification and there will be a pop-up with the option to RSVP.

There is a “create event” button exclusive to group admins and to publish the event, there is a button that says “broadcast”.

**Post-Questions:**

Considering that the list of students who RSVP an event is visible to the event facilitator/admin, can the group admin also see who scanned the QR code or is that data kept privately simply for virtual award distribution?

Are group admins expected to provide tangible rewards based on attendance?

What is the process exactly of how a student RSVPs an event through the app?

Jimmy is the president of the Astronomy club at UCI and the on-boarded admin of the astronomy group on AnteaterGroups. During his weekly club meeting, Jimmy makes plans to have a stargazing event this upcoming Friday night and have it open to all UCI students. He opens up AnteaterGroups on his phone and creates his event for the astronomy group, marking it as public and broadcasts the event to other students on AnteaterGroups.

Caitlin, a 1st year UCI student and member of the Astronomy group, sees the stargazing event in her notification tab on AnteaterGroups. Interested in attending, she clicks on the event notification, sees a pop-up with the option to RSVP, and RSVPs. As Friday night comes along, Caitlin comes to the event location and scans the QR code Jimmy has printed so that she can get her attendance logged. The event ended past midnight so Caitlin feels uncomfortable walking back to her dorm at this time, so she opens up AnteaterGroups and finds the number for the UCI Health Safety Escorts on the campus resources tab. She calls it and is soon safely escorted home.

**Scenario 4 - Friend searching and suggestion**

**By:** Nathan Van

**Source:** Field Notes #63, #70, #79

**Assumptions:** Students get friend suggestions based on their areas of interest and groups that they are associated with. The number of mutual friends will decrease if people unfriend each other.

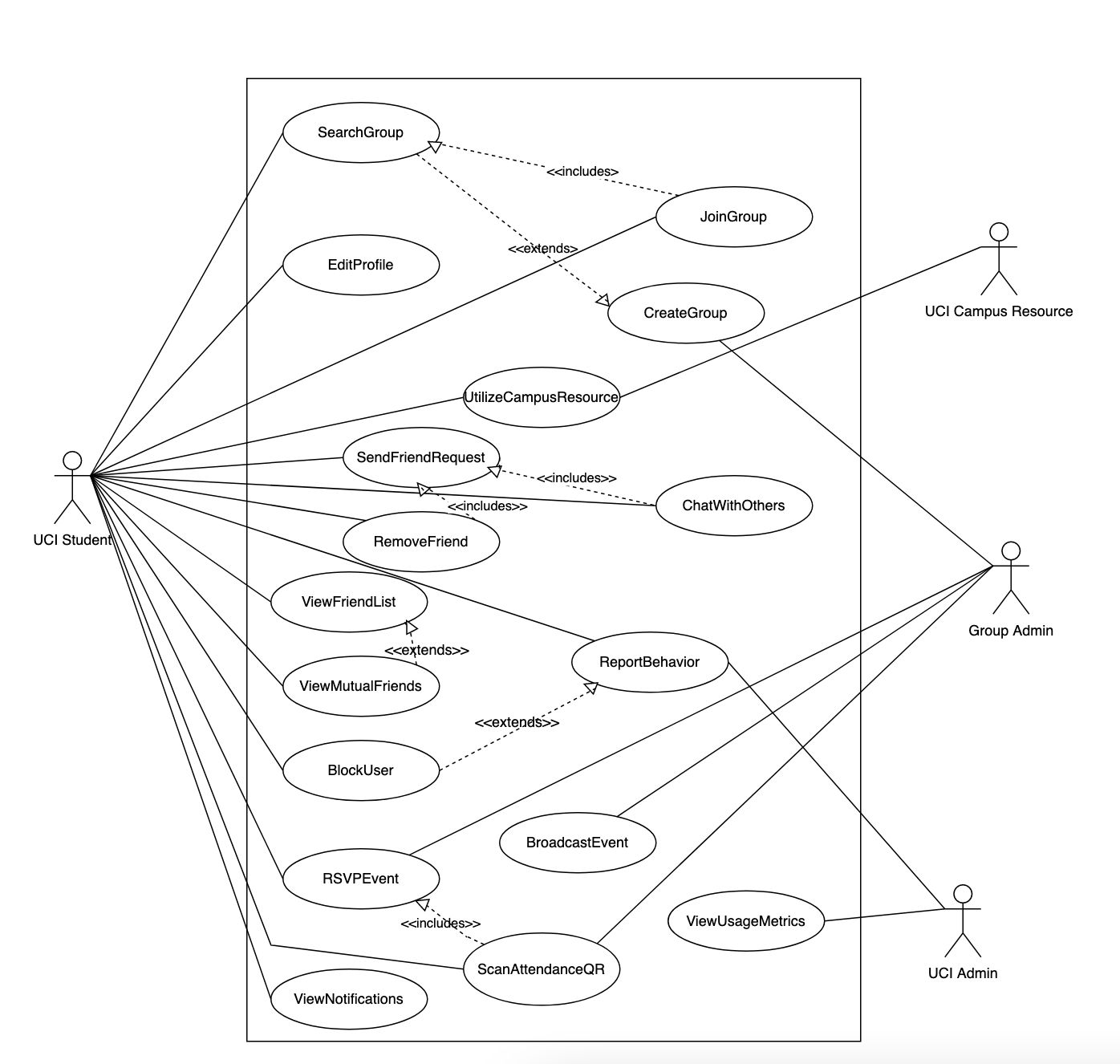
**Post-Questions:** Can students search for friends outside of their circle of mutual friends?

What are examples of areas of interest that play into the friend recommendation algorithm or system? How are friend recommendations ranked in a search?

Katherine is a UCI student who just learned about the AnteaterGroups app and downloaded it. She knows that her other UCI friends are also using it and already had their accounts on AnteaterGroups. She proceeds to search for a few names of her friends on the search bar, and right away, a list of UCI students with similar names appears on her screen. Katherine then scrolls through the list and finds her friend, and then sends a friend request to that right person.

Robert is a UCI student and Katherine’s friend at UCI. Robert receives a notification on his phone from the AnteaterGroups app notifying him that there’s a new friend request sent to him. He proceeds to open the app and sees Katherine’s friend request. Robert then accepts Katherine’s friend request and is now able to browse through her profile information and see mutual friends. Upon accepting Katherine’s friend request, Robert notices that he frequently gets friend suggestions for people with Katherine as a mutual friend.

# **Part 2.1: Use Case Diagram**



# 

# **Part 2.2: Use Case Description**

# 

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | EditProfile |
| Author | Michael |
| Priority | Medium |
| Source | Field notes 37, 44, 66b |
| Short Description | Student can edit their profile by adding a profile picture, short description,  contact info, social media accounts, and other fields. |
| Goal(s) | Enhance students’ interests.  Share certain profile information. |
| Primary Actor | UCI Student |
| Secondary Actors | None |
| Preconditions | The student must have an account registered already and is currently logged into their account. |
| Success End Condition | The student will have the newly updated information added to their profile. |
| Failed End Condition | The student will not have any new information added to their profile. |
| Trigger | The student selects an “Edit profile” option. |
| Basic Flow (Main Success Scenario) | 1. The student logs onto the application. 2. The student navigates to the About me profile section. 3. The student selects the “Edit profile” option. 4. The student views all the fields that they can update. 5. The student adds that they “enjoy swimming” in their short description. 6. The student selects an option to “save” all the changed information. 7. The student’s profile now has the updated information. |
| Alternative Flows | 1. The student logs onto the application. 2. The student navigates to the About me profile section. 3. The student selects the “Edit profile” option. 4. The student views all the fields that they can update. 5. The student adds their email to their profile. 6. The student selects an option to “save” all the changed information. 7. The student’s profile now has the updated information. |
| Exception Flows | 1. The student logs onto the application. 2. The student navigates to the About me profile section. 3. The student selects the “Edit profile” option. 4. The student views all the fields that they can update. 5. The student selects to change their profile picture but does not select one. 6. The student selects an option to “save” all the changed information. 7. The student’s profile now has the old saved profile information because they cannot not have a profile picture. |
| Relationship to other use cases | None |
| Supplementary Information | Only the name and profile picture on a student’s profile is required to fill in. A short description, contact info, social media accounts, and other fields are optional. |
| Open Issues | None |

# 

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | SearchGroup |
| Author | Michael |
| Priority | High |
| Source | Field notes 4, 45 |
| Short Description | The student will be able to search and find groups that they have an  interest in. |
| Goal(s) | Access to groups.  Enhance students’ interests. |
| Primary Actor | UCI Student |
| Secondary Actors | None |
| Preconditions | The student must have an account registered already and is currently logged into their account. |
| Success End Condition | The student will be able to view a collection of groups based on the search  terms that the student enters. |
| Failed End Condition | The student does not get to view a collection of groups that the student  searches for. |
| Trigger | The student must select the search bar, enter keywords, and submit. |
| Basic Flow (Main Success Scenario) | 1. The student navigates to the search bar and selects it. 2. The student enters “rock-climbing” and hits enter. 3. A collection of groups that are related to “rock-climbing” appear on the page. |
| Alternative Flows | 1. The student navigates to the search bar and selects it. 2. The student does not type a group to search and hits enter. 3. A collection of groups that are related to the student’s interests are shown on the page. |
| Exception Flows | 1. The student navigates to the search bar and selects it. 2. The student enters “rock-climbing” and hits enter. 3. Rock climbing is not a group so the student cannot find a rock climbing group. |
| Relationship to other use cases | Included in JoinGroup because students must search for a group before they can join it. |
| Supplementary Information | Students can only view groups that are public. |
| Open Issues | It is unclear if students are given the option to create a group if they find that a group they are interested in does not exist. |

# 

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | JoinGroup |
| Author | Michael |
| Priority | High |
| Source | Field notes 6 |
| Short Description | Students will be able to join groups that they wish to be a part of. |
| Goal(s) | Enhance students’ interests.  Join groups.  Suggest groups.  Find mutual friends.  Encourage interaction with other students.  Access to groups. |
| Primary Actor | UCI Student |
| Secondary Actors | None |
| Preconditions | The student must have an account registered already and is currently logged into their account. |
| Success End Condition | The student will be able to join a group that they want to be in. |
| Failed End Condition | The student will not be able to join a group they want to be in. |
| Trigger | The student finds a group that they are interested in joining and selects the “join” option to become a member. |
| Basic Flow (Main Success Scenario) | 1. The student is viewing a collection of groups that they want to participate in. 2. The student selects a public group. 3. The student selects the “join” option and gets added as a member. |
| Alternative Flows | 1. The student selects a group that is marked private. 2. The student selects an option “request to join.” 3. The student must wait for their request to be approved to become a member of that group. |
| Exception Flows | 1. The student is viewing a collection of groups that they want to participate in. 2. The student selects a public group. 3. The student selects the “join” option. 4. Due to a technical error the student is not added as a member. |
| Relationship to other use cases | Includes SearchGroup because students must find the group first in order to join it. |
| Supplementary Information | If a student wants to join a private group, they must request to join. Otherwise, they can immediately join a group if it is public. |
| Open Issues | None |

# 

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | SendFriendRequest |
| Author | Michael |
| Priority | Medium |
| Source | Field notes 27, 49, 67, 70, 79 |
| Short Description | Students will be able to establish a network of friends by adding other  students to their friends list. |
| Goal(s) | Improve students’ social connectedness.  “Friends” feature.  Add students’ to friends’ list. |
| Primary Actor | UCI Student |
| Secondary Actors | N/A |
| Preconditions | The student must have an account registered already, is currently logged into their account, and is on the profile of another student. |
| Success End Condition | The student is able to add other students and have them added to their list of friends. |
| Failed End Condition | The student will not be able to have a collection of friends. |
| Trigger | The student clicks on an “add friend” option on another student’s profile. |
| Basic Flow (Main Success Scenario) | 1. The student searches for a friend in the search bar. 2. The student selects the friend they were looking for. 3. The student selects the “add friend” option on the other student’s profile. 4. The friend request is sent to the other student for approval or denial. |
| Alternative Flows | 1. The student is viewing a friend’s friend’s list. 2. They select the student and enter their profile. 3. They select the “add friend” option to request them as their friend. 4. They wait for the other student to accept or decline the friend request. |
| Exception Flows | 1. Student B blocked student A. 2. Student A searches for student B in the search bar. 3. Student A does not see an “add friend” option on their profile. |
| Relationship to other use cases | Included in RemoveFriend because a friend cannot be removed if they were not added initially. |
| Supplementary Information | None |
| Open Issues | It is unclear if the student can see a student if they are blocked. |

# 

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ViewUsageMetrics |
| Author | Michael |
| Priority | High |
| Source | Field notes 38, 50 and case study |
| Short Description | UCI administration will be able to view group metrics and other usage  information related to the application. |
| Goal(s) | Increase administration/wellness center understanding of students’ behaviors.  Manage students’ app activity.  Collect group headcount.  View group participation. |
| Primary Actor | UCI Administration |
| Secondary Actors | UCI Student |
| Preconditions | The user must be a UCI administration and have an account registered with admin privileges. |
| Success End Condition | The UCI administration will be able to view all usage metrics of the application. |
| Failed End Condition | The UCI administration will not be able to view all usage metrics of the application. |
| Trigger | UCI admin selects the dashboard and the page is filled with all usage metrics. |
| Basic Flow (Main Success Scenario) | 1. The UCI administration logs into the application. 2. They select the dashboard page. 3. The dashboard page shows all the usage metrics of the application. |
| Alternative Flows | 1. The UCI administration logs into the application. 2. They select a specific group. 3. They are able to view data (headcount, active users, number of events) on the group they selected. |
| Exception Flows | 1. The UCI administration logs into the application. 2. They select the dashboard page. 3. The dashboard page does not load and UCI admin is unable to view the data due to a technical issue. |
| Relationship to other use cases | N/A |
| Supplementary Information | To prevent privacy issues associating students’ names to the groups they are in, UCI Administration can only view the number of students in a group, the number of logins, etc., and not the names. |
| Open Issues | UCI administration can view specific group metrics by selecting a particular group. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ReportBehavior |
| Author | Jaime |
| Priority | High |
| Source | Field Notes 12, 25, 29, 52, 69 |
| Short Description | Students may report behavior of other students who are not following  UCI’s Code of Student Conduct. These reports will go straight to  UCI Administration who will investigate the reports and potentially  UCI law enforcement if needed. |
| Goal(s) | Provide confidentiality settings  Maximize safety  Maximize individuals’ safety and security |
| Primary Actor | Student |
| Secondary Actors | UCI Admin |
| Preconditions | Student is in the same group as the reported student and/or the student is friends with the reported student. |
| Success End Condition | The offending student will either be warned or removed/banned from the group or possibly AnteaterGroups as a whole. |
| Failed End Condition | The offending student will not have any action taken against them by UCI Administration. |
| Trigger | The student clicks “report” on the offending student’s profile. |
| Basic Flow (Main Success Scenario) | 1. Student goes to a misbehaving student’s profile and clicks “report.” 2. The reporting student will fill out a short form specifying why they wish to report another student. 3. The UCI Administration will view the report. 4. The student is reprimanded and/or properly warned. |
| Alternative Flows | 1. Student goes to a misbehaving student’s profile and clicks “report.” 2. The reporting student will fill out a short form specifying why they wish to report another student. 3. The UCI Administration will view the report. 4. The UCI Administration takes the case to law enforcement. 5. Law enforcement opens an investigation. 6. The student is taken into questioning where law enforcement deals with the outcome of the student. 7. The student is removed from AnteaterGroups. |
| Exception Flows | 1. Student goes to a misbehaving student’s profile and clicks “report.” 2. The reporting student will fill out a short form specifying why they wish to report another student. 3. The UCI Administration will view the report. 4. The UCI Administration deems the report as not compelling enough to take action and throws the report away. |
| Relationship to other use cases | Extends from BlockUser, as students will typically block other users who display unethical behavior. |
| Supplementary Information | All reports are student-made. |
| Open Issues | It is unclear whether the UCI Administration gets back to the reporter about if the issue is resolved or not. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | BlockUser |
| Author | Jaime |
| Priority | Medium |
| Source | Field Note 25 |
| Short Description | Students may block other students if they do not wish to interact (see,  speak to) them. |
| Goal(s) | Provide confidentiality settings  Maximize safety  Maximize individuals’ safety and security |
| Primary Actor | Student |
| Secondary Actors | N/A |
| Preconditions | The UCI student must have an AnteaterGroups account. Typically, the blocked student is also in the same group as the user. |
| Success End Condition | The UCI student will have another student blocked, and will not be able to interact with the blocked student. |
| Failed End Condition | The student will not be able to block other users. |
| Trigger | The student clicks “block” on the desired student’s profile. |
| Basic Flow (Main Success Scenario) | 1. The student (user) clicks “block” on another student’s profile. 2. The blocked student’s messages are no longer visible to the user, and the blocked student can no longer contact or view the user. |
| Alternative Flows | 1. The student (user) clicks “block” on another student’s profile. 2. The blocked student’s messages are no longer visible to the user, and the blocked student can no longer contact or view the user. 3. The user changes their mind and unblocks the blocked student by clicking the “unblock” (same button as blocked but text is now changed to “unblock”) button. |
| Exception Flows | 1. The student (user) clicks “block” on another student’s profile. 2. The block is unable to go through, and the user cannot block another student’s profile. |
| Relationship to other use cases | Extends to ReportBehavior, as students may report other users who display unethical behavior. |
| Supplementary Information | Students may block and unblock other students at any time.  There are no restrictions on blocking another student. |
| Open Issues | It is not exactly clear how much the user is able to access or view the blocked student’s activity. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | RemoveFriend |
| Author | Jaime |
| Priority | Medium |
| Source | Field Note 49 |
| Short Description | Students may unfriend other users if they are already established as part  of their friends list. |
| Goal(s) | Improve students’ social connectedness.  “Friends” feature.  Add/remove students to friends’ list.  Provide confidentiality settings |
| Primary Actor | Student |
| Secondary Actors | N/A |
| Preconditions | The UCI student must have an AnteaterGroups account and is logged in. |
| Success End Condition | The UCI student will have another student removed from their friends’ list. |
| Failed End Condition | The student is not able to have another student removed from their friends’ list. |
| Trigger | The student clicks “remove friend” (same button as “add friend” but the text changes once the friend is added). |
| Basic Flow (Main Success Scenario) | 1. The student searches for the friend in the search bar. 2. The student clicks on the friend’s profile. 3. The student clicks on the “remove friend” on the other student’s profile. 4. The two students are no longer on each others’ friends lists. |
| Option Alternative Flows | 1. The student searches for the other student in the search bar. 2. The student clicks on the other student’s profile. 3. The student realizes that they are not friends, either because they were never on each other’s friend’s lists, or the other student had already removed them from their friend’s list. |
| Exception Flows | 1. Student B blocked Student A. 2. Student A searches for Student B in the search bar. 3. Student A is unable to find Student B and thus cannot remove Student B from their friend’s list. |
| Relationship to other use cases | Included in SendFriendRequest, as it is not possible to remove a student if they are not already friends. |
| Supplementary Information | N/A |
| Open Issues | It is unclear if the student can see another student if they are blocked. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ChatWithOthers |
| Author | Jaime |
| Priority | High |
| Source | Field Notes 33 |
| Short Description | Students may chat with other students if they have them added to their  friends’ list. |
| Goal(s) | “Friends” feature  Encourage interaction with other students  Improve students’ social connectedness |
| Primary Actor | Students |
| Secondary Actors | N/A |
| Preconditions | Students must already be friends with each other. |
| Success End Condition | The student is able to directly message their friend. |
| Failed End Condition | The student is unable to directly message their friend. |
| Trigger | The student clicks on the “message” button on their friend’s profile to message/chat with them. |
| Basic Flow (Main Success Scenario) | 1. Student A is already friends with Student B. 2. Student A searches for Student B’s profile. 3. Student A goes to Student B’s profile and clicks on “message.” 4. A chatroom window is displayed on the screen, and Student A is now able to send messages to Student B. |
| Alternative Flows | 1. Student A is already friends with Student B. 2. Student A is viewing their friend’s list. 3. Student A finds Student B’s profile on the friend’s list and clicks on “message.” 4. A chatroom window is displayed on the screen, and Student A is now able to send messages to Student B. |
| Exception Flows | 1. Student A searches for Student B’s profile. 2. Student A notices that Student B is not friends with Student A. 3. Student A is unable to message Student B until they are friends. 4. Student A sends a friend request to Student B in hopes of being able to message them. |
| Relationship to other use cases | Included in SendFriendRequest, as a student is unable to chat with another student if they are not already on each others’ friends list. |
| Supplementary Information | Students may chat with any other student in groups, but cannot directly message them in a one-to-one chat room unless they are already friends. |
| Open Issues | N/A |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | BroadcastEvent |
| Author | Gail |
| Priority | Medium |
| Source | Field notes 6, Goal model “Improve students’ social connectedness” |
| Short Description | Group admins have permissions to broadcast events for their groups and have the ability to mark them as public (everyone can attend), or private (must request to attend). |
| Goal(s) | Encourage interaction with other students.  Establish in-person events and meetings. |
| Primary Actor | Group Admin |
| Secondary Actors | UCI Student |
| Preconditions | Event creators must have group admin privileges. |
| Success End Condition | Group admins have their event advertised to gain students’ attention and participation. |
| Failed End Condition | Students will not see a new event in their notifications. |
| Trigger | The group admin clicks on the “create event” button. |
| Basic Flow (Main Success Scenario) | 1. Group admin clicks on the “create event” button within their group. 2. Group admin fills out the event details. Marking it as public or private. 3. Group admin clicks “broadcast” to send out the event notification to group members. |
| Alternative Flows | 1. Group admin opens up the app 2. Group admin clicks on the “create event” button on their main screen outside of the group. 3. Group admin selects which group the event is for. 4. Group admin fills out the event details. Marking it as public or private. 5. Group admin clicks “broadcast” to send out the event notification to group members. |
| Exception Flows | 1. Group admin clicks on the “create event” button within their group. 2. Group admin fills out the event details. 3. Group admin does not mark the event as public or private. 4. Group admin clicks “broadcast” to send out the event notification to group members. 5. A pop up window asks the group admin to select whether they want the event to be private or public. |
| Relationship to other use cases | N/A |
| Supplementary Information | The events being made are formally created in main groups and cannot be created in smaller subgroups. |
| Open Issues | N/A |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | RSVPEvent |
| Author | Gail |
| Priority | Low |
| Source | Field notes 74, 81, Week 7 Field notes 2, 4 |
| Short Description | Students can RSVP for an event they are interested in going to. |
| Goal(s) | Encourage participation. |
| Primary Actor | UCI Student |
| Secondary Actors | Group admin |
| Preconditions | The student is a registered member of AnteaterGroups and is a member of the group of which the event is hosted by. |
| Success End Condition | The student is marked as planning to attend the event. |
| Failed End Condition | The student can not indicate whether they are planning to attend the event. |
| Trigger | The student clicks on a broadcasted event in their notifications tab. |
| Basic Flow (Main Success Scenario) | 1. The student clicks on a broadcasted event in their notifications tab. 2. The student selects “Yes RSVP” on the pop-up option regarding RSVP-ing. |
| Alternative Flows | 1. The student navigates to a group they are a member of. 2. The student sees the events being broadcasted in the group information window. 3. The student clicks on the event they are interested in attending. 4. The student selects “Yes RSVP” on the pop-up option regarding RSVP-ing. |
| Exception Flows | 1. Non-registered student with view-only privileges sees an event that a group is having. 2. The student is unable to RSVP for the event because they are not a registered member of AnteaterGroups. 3. Student registers for AnteaterGroups with their UCINetID. 4. The student joins the group which is holding the event they are interested in. 5. The student clicks on the event they are interested in attending. 6. The student selects “Yes RSVP” on the pop-up option regarding RSVP-ing. |
| Relationship to other use cases | RSVPEvent is included by ScanAttendanceQR because someone must RSVP an event before attending. |
| Supplementary Information | Only registered members of AnteaterGroups can RSVP for events. |
| Open Issues | N/A |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ScanAttendenceQR |
| Author | Gail |
| Priority | Medium |
| Source | Field notes 74, 75, 77,  Case study: “The ability to offer and award incentives for students to participate in the community through the app” |
| Short Description | Concise description of the use case (approximately 1-3 sentences) |
| Goal(s) | Encourage participation.  Offer incentives.  Go to events. |
| Primary Actor | UCI Student |
| Secondary Actors | Group Admin |
| Preconditions | The student RSVPed for the event. |
| Success End Condition | The student is marked for their attendance and gained points towards receiving virtual badges on their profile. |
| Failed End Condition | The student does not have attendance points and cannot receive a virtual badge. |
| Trigger | Student goes to an event. |
| Basic Flow (Main Success Scenario) | 1. The student RSVPs for an event. 2. The student goes to the event, whether virtually or in person. 3. The student scans the provided QR code from the group admin and receives attendance points on their account. |
| Alternative Flows | 1. The student goes to an event without RSVP-ing. 2. At the event, the student RSVPs. 3. The student scans the provided QR code from the group admin and receives attendance points on their account. |
| Exception Flows | 1. The student RSVPs for an event. 2. The student goes to the event, whether virtually or in person. |
| Relationship to other use cases | Includes RSVPEvent because a student must have RSVPed for an event before being able to scan the event’s QR code for attendance. |
| Supplementary Information | Group admins are expected to print out or provide the QR codes for the group members to scan. |
| Open Issues | It is unclear how the data of which/how many students scanned the QR code is stored. Ensuring that the QR code is not shared with students who do not attend events has not been discussed. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | UtilizeCampusResources |
| Author | Gail |
| Priority | Low |
| Source | Field notes 26, 52, 61 and Goal model “Improve student involvement with campus resources and organizations” |
| Short Description | Students can utilize campus resources that are not run by AnteaterGroups such as UCI Health Safety Escorts, UCI Library study rooms, and UCI Wellness mental health specialists. They are able to access these resources within AnteaterGroups, either through provided contact information, library API integration, and through the ability to make appointments with Wellness Center specialists. |
| Goal(s) | Improve student involvement with campus resources and organizations.  All subgoals within the above higher level goal of goal model 5. |
| Primary Actor | UCI Student |
| Secondary Actors | UCI Campus Resource |
| Preconditions | The student is interested in getting assistance from an escort, mental health specialist, or wants to book a library study room without having to browse the web for the library website. |
| Success End Condition | The student utilized a campus resource through using AnteaterGroups. |
| Failed End Condition | The student does not take advantage of an available campus resource through using AnteaterGroups. |
| Trigger | The student navigates to the “Campus Resources” window in AnteaterGroups. |
| Basic Flow (Main Success Scenario) | 1. The student navigates to the “Campus resources” window in AnteaterGroups. 2. The student selects which resource they are interested in using.    1. If escort: Student views the UCI Health Safety Escorts phone number and has the option to dial it.    2. If library study room: Student has the option to choose to book an available study room.    3. If meeting with a mental health specialist: Student has the option to make a Zoom appointment with a mental health specialist. |
| Alternative Flows | 1. The student navigates to the “Campus resources” window in AnteaterGroups. 2. The student selects which resource they are interested in using.    1. If escort: Student views the UCI Health Safety Escorts phone number and has the option to dial it.    2. If library study room: Student has the option to choose to book an available study room.       1. The student only views available study rooms on the app and later proceeds to the UCI Library website to book the room.    3. If meeting with a mental health specialist: The student has the option to make a Zoom appointment with a mental health specialist. |
| Exception Flows | 1. The student navigates to the “Campus resources” window in AnteaterGroups. 2. The student selects which resource they are interested in using.    1. If escort: Student views the UCI Health Safety Escorts phone number and has the option to dial it.    2. If library study room: There are no available study rooms. The student is unable to book a room at this time.    3. If meeting with a mental health specialist: There are no available appointments for when the student is available. The student is unable to make an appointment at this time. |
| Relationship to other use cases | N/A |
| Supplementary Information | Students are welcome to use AnteaterGroups to access these resources, however they are not limited to the app only to access these resources. Students can continue to access campus resources through their respective offices and websites. |
| Open Issues | It is not confirmed whether UCI libraries and UCI mental health specialists have agreed upon working with AnteaterGroups to provide these integrated features. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ViewMutualFriends |
| Author | Nathan |
| Priority | Medium |
| Source | Field Notes 27, 70, 79 |
| Short Description | The student are able to see mutual friends over the person’s profile/friend list  and/or list of members in a group. |
| Goal(s) | Improve students’ social connectedness.  Encourage interaction with other students.  “Friends” feature.  Suggest mutual friends. |
| Primary Actor | The UCI Student. |
| Secondary Actors | N/A |
| Preconditions | The student must create an AnteaterGroups account and have at least one friend in their friends list. The mutual friend’s friends list is set to public and not hidden from the public. |
| Success End Condition | The student are able to see mutual friends when browsing over other people’s  friend lists or the list of members within a group. |
| Failed End Condition | The student does not see any mutual friends on other people’s friend lists or within a group. |
| Trigger | The student browses through friend lists or searches for friend names. |
| Basic Flow (Main Success Scenario) | 1. The student opens their AnteaterGroups app. 2. Student searches for a friend's name. 3. Student sees a list of people with similar names. 4. The student sees a number of mutual friends on each profile info. 5. Suggestions rank by the number of mutual friends a person has. 6. The student searches for a group. 7. Student sees a list of mutual friends in the group. |
| Alternative Flows | 1. The student selects options to have their mutual friends display or not. 2. The student enables/disables search rankings by mutual friend or not. 3. Groups hidden its list of members, not allowing The student to see mutual friends within the group. |
| Exception Flows | 1. The student searches for a UCI friend on AnteaterGroups but none of the queried names match with the person they’re looking for. 2. The student knows that their UCI friends also joined the same groups as them but could not find them due to their UCI friends disable the mutual friend search in the setting. 3. The search system displays the list of queried names not correctly ranked. |
| Relationship to other use cases | Extends ViewFriendList because a student can find mutual friends from another student’s friend list, but they may also find mutual friends otherwise from suggestions. |
| Supplementary Information | Certain privacy settings disable the ability for the student to search a particular individual and/or hide the list of mutual friends from name searches. The student excludes their profile from name searches, and their name won’t be displayed when their UCI friends search for them. |
| Open Issues | N/A |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ViewFriendList |
| Author | Nathan |
| Priority | Medium |
| Source | Field Note 49 |
| Short Description | The student have a friends list on their profile that can be viewed by them  and/or their friends. |
| Goal(s) | Improve students’ social connectedness.  Encourage interaction with other students.  “Friends” feature. |
| Primary Actor | The UCI Student. |
| Secondary Actors | N/A |
| Preconditions | The student must create an AnteaterGroups account and have at least one friend in their friends list. Friends list is set to public and not hidden. |
| Success End Condition | The student is able to view their friends list and/or view their friends’ friends list. |
| Failed End Condition | The student is unable to view their friends list and/or their friends’ friends list. |
| Trigger | The student click on their friends list or their friends click on their friends list  on their profile. |
| Basic Flow (Main Success Scenario) | 1. The student opens their AnteaterGroups app. 2. The student clicks on their profile. 3. The student views their friends list on their profile. 4. The student browses their friends on their friends list. 5. The student clicks on a friend and browses their profile. 6. The student clicks on the friend’s friends list and sees a list of people. 7. The student searches for a friend’s name on the search bar. 8. The student sees the person they’re looking for and clicks on his/her profile. 9. The student accesses the person’s profile and sees their friends list that is set to public viewing. |
| Alternative Flows | 1. The student opens their AnteaterGroups app. 2. The student accesses his/her settings and disable public viewing on   his/her friends list.   1. The student’s friends open their settings and disable their friends list from public view. |
| Exception Flows | 1. The student opens their AnteaterGroups app. 2. The student clicks on their profile and views their friends list. 3. The student clicks on one of their friends but cannot view the friend’s friends list 4. The student searches for a friend’s name and clicks on his/her profile. 5. The student couldn’t view the friend’s friends list due to that friend disabled public viewing. |
| Relationship to other use cases | ViewFriendList is extended by ViewMutualFriends because mutual friends can be found from a friends list. |
| Supplementary Information | Certain privacy settings disable the ability for the student to view a particular individual’s or friend’s friends list. The student can exclude their friends list from public viewing, and this prevents his/her friends from accessing the student’s friends list. |
| Open Issues | N/A |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | CreateGroup |
| Author | Nathan |
| Priority | High |
| Source | Field Notes 5, 6, 10, 13, 39, 48, 63, 65 |
| Short Description | UCI Students, Already Registered UCI Clubs/Organizations, and Event  Facilitators like counselors can create groups on the AnteaterGroups app.  The capacity limit is set by the group admin at the beginning of the group  formation stage. Duplicate groups get notified when found. |
| Goal(s) | Join groups.  Improve students’ social connectedness.  Encourage interaction with other students.  Access to groups. |
| Primary Actor | The UCI Student and The Event Facilitator. |
| Secondary Actors | The UCI Admin. and The Counselor. |
| Preconditions | Group formations must meet the minimum requirements by the AnteaterGroups app and follow the AnteaterGroups’ guidelines for authorized topics and materials. There must be an admin for every group. |
| Success End Condition | The group meets the minimum requirements set by the AnteaterGroups app and is able to be joined by other students. Joining groups either needs a passcode or not, which is set by the admin. |
| Failed End Condition | The group fails to meet the minimum requirements set by the AnteaterGroups app and is unable to be joined by other students. The group is automatically deleted by the AnteaterGroups app. |
| Trigger | The actor(s) click on the CreateGroup icon when you’re logged in and fill out the required blank fills or sections laid out by the AnteaterGroups app. |
| Basic Flow (Main Success Scenario) | 1. The actor(s) opens their AnteaterGroups app. 2. The actor(s) clicks on the CreateGroup icon. 3. The actor(s) fills out the required fields. 4. The actor(s) sets public or private. 5. The actor(s) creates an access code for private groups. |
| Alternative Flows | 1. The actor(s) opens their AnteaterGroups app. 2. The actor(s) clicks on the CreateGroup icon. 3. The actor(s) selects imports from existing UCI clubs/organizations. 4. The system sends a request to the appropriate admin for permission to replicate their groups on the AnteaterGroups. 5. The admin of the UCI clubs/organizations permits the replication of their groups. 6. The groups are created with the settings similar to those found on the UCI clubs/organizations. |
| Exception Flows | 1. The actor(s) opens their AnteaterGroups app. 2. The actor(s) clicks on the CreateGroup icon. 3. The actor(s) fills out the required fields. 4. The actor(s) sets public or private. 5. The system notifies the actor(s) that there’s a duplicate group. 6. The system allows the actor(s) to choose to proceed anyways or cancel. 7. The actor(s) clicks cancel and the app is back to the home screen. |
| Relationship to other use cases | CreateGroup is extended by SearchGroup, because if a group is not found when a student searches for it, then they have the option to create that group. |
| Supplementary Information | There’s many customizations to CreateGroup where the actor(s) can choose to enable/disable those options for their group(s). These features include the ability to RSVP for events from groups and scan QR code for attendance in groups. |
| Open Issues | It is unclear the negative impacts from allowing all UCI students to be able to create groups. Making sure that group formations are ethical and do not violate any UCI policies. |

| **Section** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ViewNotifications |
| Author | Nathan |
| Priority | Medium |
| Source | Field Notes 18, 41, 68, 86 |
| Short Description | Students and other administrative bodies are able to receive notifications for certain areas in the AnteaterGroups app. A list of notifications are group in a notification bell located on the top right hand corner of the home screen and can be accessed through clicking on it. Notification options can be set from the settings to allow pop up on lock screens or not. |
| Goal(s) | Enhance students’ interests.  Encourage interaction with other students.  Suggest mutual friends. |
| Primary Actor | The UCI Student. |
| Secondary Actors | The UCI Admin., the UCI Counselor, and the Event Facilitator. |
| Preconditions | The actor(s) must have their notification feature enabled in the settings. |
| Success End Condition | The actor(s) clicks on the notification bell on the top right hand corner of their home screen and sees a list of notifications ranging from friend requests, event notifications, messages to maintenance announcements. |
| Failed End Condition | The actor(s) clicks on the notification bell on the top right hand corner of their home screen and does not see any notifications generated by the app. |
| Trigger | The actor(s) clicks on the notification bell on their home screen. |
| Basic Flow (Main Success Scenario) | 1. The actor(s) opens their AnteaterGroups app. 2. The actor(s) clicks on the notification bell. 3. The actor(s) sees a list of notifications from different features on the app. 4. The actor(s) touches on one notification and gets directed to the feature that produced the notification. 5. The system deletes the notification that touched on. |
| Alternative Flows | 1. The actor(s) receives a notification for a new message in their chat. 2. The actor(s) opens the AnteaterGroups app. 3. The actor(s) goes directly to the chat box and reads the new message. 4. The system automatically deletes the notification for the unread message. |
| Exception Flows | 1. The actor(s) opens the AnteaterGroups app. 2. The actor(s) disables the notification feature for new messages. 3. The actor(s) sees a new message in their chat box but no notifications from it. |
| Relationship to other use cases | N/A |
| Supplementary Information | None |
| Open Issues | It is not clear which notification can be effective and not bothersome for the AnteaterGroups app. Making sure that notifications are displayed in a  way that can be easily seen by the UCI Student, the UCI Admin, the UCI Counselor, and the Event Facilitator. |

# 

# **Field Notes**

## **(Week 2) - 4/12**

1. 10-12 fixed groups (no strict number)
2. Ask interest /hobbies/likes upon registration
3. Customization on front page of students interest/hobbies/etc
4. Can you search for groups by some tags or by a group name?
   1. Yes, make it as user friendly as possible
   2. When the student logs in, would be good to search for a group using name or tag
      1. Names and subjects
      2. Should match keyword even if it isn’t exact (near match)
      3. Tags: category (sports), group names: specific (football, baseball etc)
5. Should students be able to promote their own groups/events?
   1. Admin is allowed to promote group or events
   2. Students can be admins of groups they create and promote group activities
   3. Afraid of giving advertisement privilege to each and every student joining in, might be a lot of noise / not a lot of precise advertisements
6. Who are allowed to create groups/events? Who gives admin level account status to students? How to manage duplicate groups/events?
   1. Student leaders need verification with the admin of the app to get permission to create groups/events.
   2. Admin accounts can set groups/events as public (everyone can join) or private (request to join).
7. What are the ways to offer and award incentives to students? Do students get digital goods, physical goods, or reward points to be redeemed through the app or through a third-party partner?
   1. Students can participate in events and get reward points which can then be used to purchase digital goods and digital gifts.
   2. Digital goods and gifts can be sent from student to student.
8. Students who manage roles (ex. ARC: gym, swimming pool) can be onboarded as admins by default
   1. Campus admin would pick a student to be an admin of each group
9. How would rewards be distributed?
   1. Reward based on attendance
      1. Example: yoga meditation, simple reward system, if the student has more than 50% attendance, he/she gets some reward which can be given by the person conducting yoga sessions
   2. Need some system (db) to track attendance
   3. Gold/silver badges based on attendance and will be visible on ‘About Me’ section in profile
10. Multiple groups for the same subject?
    1. Students have the ability to create private groups within bigger group
11. Where can students form groups? On-campus only or any locations off-campus is fine too? Can they form groups at private locations?
    1. Due to the safety and security of the students.
    2. Campus Safety Council, Public Safety, Police Department
12. Who will get reports on users’ interactions and violations?
    1. To allow abusive/inappropriate behavior be reported to the student safety council and the police if necessary. To avoid student bullies and promote professionalism.
13. Organizations and clubs that have already been established - will they be able to create groups so they can recruit students?
    1. Yes (?) board all orgs on the app
14. If we see a class that is taught by 2 diff professors, students might want to have a group specified for each professor
    1. Repeating groups is fine with overlapping classes/different professors (making 2 groups)
    2. People like to shadow groups, doesn’t have to be restrictive, let people mix up/sign up for any groups they want
15. Potential launch date: September 2022
16. Should support at least 10k users (note: 30k students if all students participated)
    1. Should keep in mind scalability
17. Messages in specific channels can be seen by everyone regardless of their group (advertisement channel?)
18. Can send push notifications to everyone for important announcements
    1. Settings: allow user to enable or disable this
19. **Note: no alumni access allowed**
20. Groups centered around classes and sections - do these groups expire at the end of the quarter (to discourage collaboration)?:
    1. We don’t really need to onboard the classes as groups in our app
    2. App is used to promote connectivity, already have canvas for classes, redundant
21. Don’t include academic classes (what you see on webreg or canvas)
    1. Academic clubs are fine
22. How do we gather data on students’ interest to suggest groups?
    1. Competitor app: MeetUp (borrow registration process)
23. Can there be groups for certain majors or schools but not specific courses?
    1. Yes
24. Both android and apple compatible
25. Will the admin check for academic dishonesty or will students have privacy within their groups?
    1. Detect inappropriate/abusive chat with machine learning algorithm, proceed with warning then block student from using application if behavior continues 2 or 3 more times
    2. Students will also have the ability to report other students if there is inappropriate behavior and admin will be able to review report and take action
26. What other features besides creating and promoting connectedness and personal growth should the app have? Do students have the option to seek a mental health hotline or reach a specialist to address their mental health issues through the app?
    1. To further address and enhance social healthiness among the students.
    2. UCI Administration, UCI Student Affairs, UCI Center for Student Wellness & Health Promotion, Software Engineers.
27. Should students see friend suggestions as well as group suggestions? (like if you have a mutual friend w/ someone, will they be suggested to you?)
    1. yes
28. Features to police the app activity for inappropriate behavior
    1. Yes, try to detect inappropriate/abusive behavior using tech first (engineering challenge)
29. When a student reports another student, does the admin come in and view the activity in the chats? Or is there any expectation of privacy, or encryption? Does the administration have full access to all communications?
    1. Don’t give admin full access to chats
    2. Reporting will go to admins along with previous 4-5 chats where the user flagged as inappropriate
30. Should applications support e2e?
    1. Yes, to ensure privacy among students and app data
31. What happens when the lead admin graduates? Will the group be deleted or expired, or will someone take over the admin role?
    1. We don’t want the group to be deleted, would also like to pass on admin access to another student
32. Will the admin role be randomly assigned?
    1. Anything is fine as long as there’s an admin in the group
33. Can we always have a private chat feature?
    1. Yes, once someone in the group is your friend you can privately message them
34. Can students access anteater groups through a website or is the website for admins only?
    1. Just the app for now
35. Where can students form groups? On-campus only or any locations off-campus is fine too? Can they form groups at private locations?
    * 1. Due to the safety and security of the students.
      2. Campus Safety Council, Public Safety, Police Department.

## 

## **(Week 3) - 4/15**

1. Priorities
   1. Prototype: should have bare minimum of login, can see groups on home screen, can customize it for each student depending on hobbies, functioning chat system,
2. Will students be able to use aliases for their profiles or should they use real names?
   1. Real name
   2. Provide as much flexibility as possible
3. Campus admin can see usage statistics - what other metrics should admin be able to view?
   1. How many students logging in/registering, centralized dashboard of how many students have created groups, single page,
   2. Consistency ?
   3. Data about performance, if groups are/aren’t performing well
      1. Can reach out to group admins to ask them how to make it more engaging
   4. Only public group data ?
4. Can students create groups on their own or do they need to reach out to administrators?
   1. Give them the freedom to make their own groups
   2. Ability to create private groups (can toggle/change at any time - maybe only group admin can change?)
   3. Private groups invisible to public
   4. Students have the option to create public/private groups (can change after creation)
   5. Let’s not have private/public groups but ‘subgroups’ where they are all visible to students but some subgroups have password secure or approval permissions by the admin.\*\*
5. Percentage of up time for application?
   1. Available 24/7
   2. What time should maintenance be performed? (discussion among engineers)
      1. Find time where app is used the least and perform maintenance time during that
6. Will students get notifications or reminders for maintenance?
   1. Once in two months or whenever there is a major break/crit issue
7. What kind of rewards should students receive for participating in group activities/events?
   1. Badges (bronze, silver, gold, etc, titles)
      1. Visible publicly
      2. For other students to see and connect with each other
      3. To get an idea of the student profile
   2. Points
8. How easy is it to integrate the library API?
   1. Libraries as stakeholders
   2. Would require looking at the code
9. What info do students need to complete their profile?
   1. Name
   2. Profile pictures
   3. Short description (optional)
   4. mail id or phone number (optional)
   5. Media upload/ social media links (optional)
10. How to encourage students to explore new interests?
    1. Initial data with their interests from registration on home page, option to see all groups or top 10-15 interests/groups, option to search for groups
11. Tags for age requirements (ex. 21+)
    1. Need to show id
12. Should the app integrate with a third-party map for locating events?
    1. third -party map integration
13. Creating duplicate groups
    * 1. If student creates a subgroup that is similar to existing, ask them if they want to
    1. still create or not
14. Friends list
    1. Close friend sublist
    2. Can create subgroup

## 

## **(Week 4) - 4/22**

1. Data visualization
   1. Admin will be able to see metrics (app usage, headcount in each group)/data on a single page.
   2. Admin can take measures if the headcount in a group is going down; promote more activities in a group.
2. Timeline
   1. Mock up: 1 month + 1 week
   2. Final product: Targeting release for the next AY Fall 2022 (September 2022).
   3. Have the application ready by orientation so that the department can market or promote app awareness. (MVP)
3. In case of inappropriate behavior, can users contact the police through the app?
   1. Workflow: users report, admins review report, then admin can contact police if needed.
   2. Possible late night concerts: have the features to take the shuttles back home.
   3. Make use of existing UCI services.
   4. Show the phone numbers ^^ on the app
4. Will there be ways to video chat with the mental health specialist?
   1. Students can schedule sessions over Zoom or Google Meet through the app
5. Will there be any accessible options/features for people with disabilities?
   1. Yes, w/ colors and fonts (will require research on what these students need)
6. Dark Mode Feature
7. What would other accounts (professor) be able to view on the application?
   1. Let the view be the same, but have some of the privileges of admins.
   2. Unique features only available to admin accounts
      1. Ex. the right to remove a student from a group
8. Will grad students have different privileges compared to undergrad students?
   1. Let them have the same privileges
9. Are faculty members allowed to join, and if so, can students connect with them like they do with their peers?
   1. no , trying to promote connectivity of students
   2. If they’re trying to train people (ex. Yoga group) they’re allowed to join(onboarded) but can’t use it like an end user
   3. Unlikely be end users
   4. Can be admins
10. Almost everyone associated with UCI has netID so we need to check the database for student status before every login.
11. What is the budget range that will be allocated for this application?
    1. I think he answered this - depends on skills of engineer and other stuff
12. How to reserve locations to meet
    1. library/study centers have their own system so we use that
    2. For non reservable places: let’s not bother because everyone one has access to aldrich park for example
    3. Have a popup reminder or show somewhere on the app that certain locations are under maintenance or are closed

## 

## **(Week 5) - 4/29**

1. What happens when a group gets too big or has been established for a long time?
   1. “Official Group of UCI”
2. Can anyone create groups or do students have to create subgroups of groups?
   1. Students can have both
   2. Already a group for things like music (100 students registered automatically)
   3. Friends list
   4. You can create a mini group/subgroup
3. Want to have pre-established groups and also groups that students establish when they have the app
4. What if a club/group at school doesn’t want to be onboarded on the app?
   1. Create the Anteater Group, but don’t make them the admin, but if nobody wants to be the admin, don’t create the group at all for them
5. Campus clubs sync with Anteater Groups?
   1. Get permission from the hosts in campus clubs if they want to join Anteater Groups or not before listing them on Anteater Groups.
   2. Can turn group profile off/on
6. Ads on the app?
   1. Group suggestions
   2. Based on close friends
      1. Ex. Your friend is part of this group, would you like to join?
   3. App get data from Google searches?
      1. -make suggestions with keywords
7. Where are notifications stored?
   1. Have a bell icon they can click on, and on that page it’ll show all their notifications
8. What happens if you report a group itself? Or the admins of a group?
   1. The report should go straight to UCI Administration
   2. Report incidents like an event not exist
9. How to make connections or send friend requests over the app?
   1. Browse the name of the person then able to send a friend request
   2. Able to see mutual friends over the person’s profile/friend list
10. Who has access to the reviews of a group?
    1. Rating out of 5 (half-steps allowed, ex. 4.5/5)
    2. Everyone can see them
    3. Separate categories (how active the group is, etc.)
    4. Only members in a group can make reviews
    5. Reviews are both for students/general public and groups
       1. Groups: to gain feedback about how they’re doing
       2. Students: to incentivize them to join the group
11. Are reviews anonymous or are they attached to your name?
    1. Not anonymous, since it could be easy to write things that hurt the group (false accusations)
    2. Reports are anonymous

## **(Week 6) - 5/6**

1. Do we want to make our own video services or use a third party service?
   * Use Zoom. (forgot what else he said)
2. The reward system based on attendance
   * We want the actual count of attendance
   * Let’s not rely on RSVP feature (may end up not coming)
   * Scan QR code (when they arrive) that tracks attendance, most accurate method
   * Keep RSVP feature to know how much to expect and prepare stuff for headcount
3. How QR system works
   * Physical: Group admin will get QR code, will be placed outside of door, attendees will scan code upon arrival
   * Online (Zoom): Provide unique keyword to enter check-in (similar to UCICheckIn)
4. UCICheckIn: Integrate if feasible and will make it easy
5. Group admins can generate QR codes on their one/ App can generate the QR code upon request.
   * Student will have to request for QR code
     + A form will populate once the QR code is scanned
6. The information for attendance
   * The name of the student
7. Friend recommendations
   * When registering for the first time: ‘these might be the friends you want to friend’
   * Mail id or contact info.
   * Compare the contact on the app with on your phone
   * Import contact info on your phone to see friends on the app with such phone #s
   * Mutual friends
   * Mutual Groups (see mutual groups like on discord??)
   * Visibility option: Seen in group or seen outside of group
8. Can pin announcement/advertisement to top of group
9. Once someone RSVP for an event, it goes to their calendar
   * People’s calendar can be visible to everyone (free slot / non-free slot)
10. Event safety concerns
    * Provide guidelines in terms of how many users can be there
    * Detect if certain events are protests/ need UCIPD as security
    * Check off boxes for the category of the event (entertainment, protest, etc.)
11. Have status of the user

* Online, offline, busy

1. Layout of groups user is in

* Groups they’re most active in at the top
* Groups in rows and columns
* Message bar also in top right corner
* About me section top left
* Each group has own notification bar

85. Common channel of notifications

* + notification button top right corner

86. Can group admins give out their own personal badges or do they choose from a select few

* Admins can’t create their own badges
* Badges can be given out automatically based off attendance, from when they scan qr codes

## **(Week 7) - 5/13**

1. Reward system:
   1. Students who attend events should get some reward points
2. Login as guest?
   1. When they want to join a group they’d need to sign up
   2. But they can see all groups
   3. View group information (all events happening)
   4. View-only privilege
   5. Security concerns?
      1. Guests can view members of group?
      2. If it sounds logical then it’s probably like that
   6. Can’t report as a guest
   7. Can’t rsvp and check in to events
3. About me section
   1. Show groups on profile (optional)
      1. If you turn it off: other people won’t see which groups you’re part of
         1. Also won’t be able to see you as a member of that group
   2. Tab: show interests on profile (also optional to make public)
4. Groups have public and group-private events
   1. Public events can be seen by view-only guests
5. Ads
   1. Small and smartly placed, so that it’s not too annoying/distracting to users
   2. Even if we get funded from UCI, we should not block income flow
   3. If it is legal, we can use it
   4. There should be some kind of regulations for choosing the ads
   5. Eventually, ads should be customized to user interests; be personalized
6. Raise tickets?
   1. To bring up their concerns/issues when using the app
   2. The tickets go to the support team who will fix the issue
   3. Chat system: to fix the issues in real time
   4. Who will be supporting this?
      1. Developers at first
      2. Later, train people to solve these kinds of requests
7. Sub-groups
   1. Just an informal version of the group
   2. Just chatting or make informal meetups with smaller groups of people
      1. Going for coffee for example
      2. Not an official creation of event
8. Covid-19 Safety
   1. Safety requirements and compliance (bringing in covid vaccinations, test results, etc.)
   2. Track students’ covid-19 test results (since linked through UCI)

**Team Meeting Minutes**

| **Team ID: 3** | | **Date: 05/14/2022** |  | |
| --- | --- | --- | --- | --- |
| **Team Members (Name)** | | **Role** | |
| 1. Gail Manlapaz | | Facilitator | |
| 1. Michael Nguyen | | Participant | |
| 1. Jaime Park | | Recorder | |
| 1. Nathan Van | | Participant | |
|  | |  | |
|  | |  | |
| **Agenda for this meeting, List of agenda items** | | **Outcomes** | |
| 1. Do homework 4. | | We finished homework 4 by the deadline. | |
| 1. Have scenarios already done individually so that we can work on the use case diagram together. | | We finished our scenarios. We all worked on the use case diagram together on diagrams.net, focusing on the actors we each used in our use cases. | |
| 1. Finish use case descriptions. | | We finished them before the deadline. | |
| 1. Put together homework 4 documents. | | We finished homework 4. | |
|  | |  | |
| **Problems encountered** | | **Resolution** | |
| 1. We set a soft deadline for Thursday night to be done with our individual portions, but all of us were set back a little. This caused us to feel a little more stressed doing the homework the day it is due. | We met up on Discord Saturday and worked together to finish the homework. | | |
| 1. We had trouble generalizing our use cases a little more so that we would not have over 20 use cases. | We tried to see from the perspective of the user and picture what are the main things they would want from the app. | | |
| 1. We had trouble understanding the directionality of the includes and extends. 2. We did not know at first what alternative flow and exception flow had to consist of. | We had to all look back at the lecture slides and make sense out of the possible scenarios to see what includes/extends what.  We had to look at the sample assignments and we saw that alternative flows are other ways to achieve a task whereas exception is if there is an issue or if the task cannot be completed. | | |
|  |  | | |
|  | |  | |
|  | |  | |
| **Plans for next meeting: Activity** | | **Responsibility** | |
| 1. Go to next elicitation meeting, | | All plans are the responsibility of all group members. | |
| 1. Ask our questions based on our assumptions for homework 4. | |  | |
| 1. Begin homework 5. | |  | |
|  | |  | |
|  | |  | |