Anteater Groups: Systems Requirements Specification

Team 3

Gail Manlapaz

Michael Nguyen

Jaime Park

Nathan Van

[**Introduction**](#_amuapadli1h5) **2**

[Purpose](#_vg34xuv3esdc) 2

[Scope](#_dql45qmxr4ro) 2

[Definition, acronyms, and abbreviations](#_z21stjuwtlbq) 3

[References](#_dzg197vavoix) 4

[**General description**](#_ekp9fw47v4tz) **4**

[Product Perspective](#_uifv298mibcv) 4

[Product Functions](#_qisyyzxmfaqq) 5

[User Characteristics](#_74eh82nx7k4c) 6

[General Constraints](#_g7fwvtjftev8) 6

[Assumptions and Dependencies](#_pt6zh5bkak83) 6

[Apportioning of Requirements](#_90kozgeqe0ho) 7

[**Specific Requirements**](#_5qifoltcdp9c) **7**

[Essential Requirements](#_r1axy1ilbmpa) 7

[Functional Requirements](#_kyklkjmi3lku) 7

[Non-functional Requirements](#_o5914yqi9cng) 14

[External Interface Requirements](#_e2qk675dl0d3) 16

[User Interfaces](#_97kux3dn7rvu) 16

[Hardware Interfaces](#_vfr0mjnsyt5p) 22

[Software Interfaces](#_mjppddaqix09) 22

[Logical Data Model](#_blhmny26ihif) 24

[**Appendix**](#_gphcmkev4o5z) **26**

[A.1.1 Stakeholder Model](#_sksn26wleibz) 26

[A.1.2 Goal-Oriented Analysis Models](#_f0kn9klj8jyg) 30

[A.1.3 Usage Model: Scenarios and Use Cases](#_9pnmn47avm4f) 45

[A.1.4 Domain Model: Class Diagram](#_c72x3ow5ufat) 80

[**Team Meeting Minutes**](#_ncquhg62teo1) **85**

[**Missing Information**](#_q78pu4gj7cs5) **86**

[**Field Notes**](#_woki9ad6ffsg) **87**

[(Week 2) - 4/12](#_v6uo1d5z4pca) 87

[(Week 3) - 4/15](#_ajxhyt8dct2) 90

[(Week 4) - 4/22](#_9q19hk2g020u) 91

[(Week 5) - 4/29](#_bc9ajs8yezwe) 92

[(Week 6) - 5/6](#_9ldo9mk8sar8) 93

[(Week 7) - 5/13](#_bkt2plehnydg) 95

[(Week 8) - 5/20](#_unmfph7hfaz6) 96

[(Week 9) - 5/27](#_yomae6xbm7t2) 96

# Introduction

## **Purpose**

Due to the COVID-19 pandemic, in-person classes at UCI have finally resumed for the 2022 academic school year. During the period of online/virtual-remote learning, many students have been unable to make friendships and connect with students and student-led organizations. Anteater Groups aims to alleviate this problem by assisting socialization among students. Students will be able to explore and develop current and new interests by joining groups full of like-minded individuals. Anteater Groups additionally aspires to improve student mental health through enriching students’ social connectedness and personal development. Both campus and student-led organizations have the ability to promote campus-wide events for students, instigating school spirit and camaraderie.

## **Scope**

The software system to be produced shall be named Anteater Groups. Anteater Groups shall be a mobile application designated for current UCI students as well as a web-based subsystem for UCI administration. The system shall allow students to join groups based around both academic and extracurricular subjects and chat within them, and promote campus activities and events to improve student social connectedness and encourage students’ exploration of interests. For UCI Administration, the app shall be able to provide usage statistics showing insight on group performance and student participation, as well as alert them of inappropriate use within the app to better improve student re-engagement with one another.

## 

AnteaterGroup’s core functionalities include:

1. Enhance Students’ Interests
   1. Offer suggestions and recommendations based on the student’s browsing history, and the groups and friends the student is associated with.
   2. Offer activities and events that can discover the student’s interests.
2. Improve Student’s Social Connectedness
   1. Offer the ability to add friends with other UCI students
3. Maximize Individuals’ Safety and Security
   1. Allow students and administrators to report inappropriate behaviors and non-affiliated groups created on the app.
   2. Allow the group admin to block, kick, or report a member for group policy violations.
4. Increase Administration/Wellness Center Understanding of Students’ Behaviors
   1. Acquire statistical data on the students’ activities on the app through report logs.
5. Improve Student Involvement with Campus Resources and Organizations
   1. Provide links to campus resources and organizations.

## **Definition, acronyms, and abbreviations**

| UCI | University of California, Irvine |
| --- | --- |
| OIT | Office of Information Technology |
| CSWHP | UCI Center for Student Wellness and Health Promotion |
| UCInetID | Electronic identification used for many online services at the University of California, Irvine. |
| Duo Mobile | Multi-factor authentication application used alongside every account using a UCInetID |
| RSVP | Response to an event invitation: attending, not attending, maybe |
| QR Code | A 2-dimensional bar code used to access information easily by scanning with the camera application on a smart device |
| iOS | Operating system used for mobile devices manufactured by Apple Inc. |
| Android | Operating system used for mobile devices manufactured by Google |
| Zoom/Google Meet | Video communications application that allows for virtual video and audio conferencing |
| API | Application programming interface; a service/contract that defines how two entities (client-server) can communicate with each other |
| ADA | Americans with Disabilities Act, a civil rights law that does not allow discrimination towards individuals with disabilities. |

## **References**

1. [UCI Health Safety Escort](https://www.police.uci.edu/how-do-i/safety-escort.php) <https://www.police.uci.edu/how-do-i/safety-escort.php>
2. [UCI Libraries](https://www.lib.uci.edu/) <https://www.lib.uci.edu/>
3. [UCI Wellness, Health & Counseling Services https://whcs.uci.edu/](https://whcs.uci.edu/)
4. [UCI Police Department](https://www.police.uci.edu/) <https://www.police.uci.edu/>
5. [Zoom](https://zoom.us/) <https://zoom.us/>
6. [Google Meet](https://meet.google.com/) <https://meet.google.com/>
7. [ADA](https://adata.org/learn-about-ada) <https://adata.org/learn-about-ada>

# General description

## **Product Perspective**

Anteater Groups will interact with UCI Administration and CSWHP through a web-based subsystem, or with undergraduate and graduate students through an application on an iOS or Android mobile device. Select UCI-affiliates may also use Anteater Groups through a mobile application as onboarded admins. Account ownership and login security shall be maintained through UCInetIDs serviced by UCI OIT. In integrating library study room booking, contact with UCI Health Safety Escorts, and making appointments with CSWHP, Anteater Groups interacts with the UCI Library API, UCI Health Safety Escorts, and CSWHP mental health specialists utilizing Zoom or Google Meets for meeting appointments. When students RSVP for an event, Anteater Groups shall log the event on students’ mobile device calendar.

## **Product Functions**

Anteater Groups shall allow students to create customizable profiles with a profile picture and a description/bio. It shall also allow students to add their peers onto their friends list, where they can then message a friend directly.

Anteater Groups shall allow students to create, search for, and join groups. Groups may be created based on academic or non-academic interests, such as a UCI course or a hobby. The software shall also allow students to create subgroups. UCI faculty members and students shall be able to be appointed as group admins. Group admins have the ability to remove a student from a group, view metrics, and control the process of creating, broadcasting, and promoting events.

Anteater Groups shall send notifications for messages as well as reminders for group events.

Anteater Groups shall allow UCI administration to view reports on usage and user engagement, create and promote groups based on campus activities, monitor activity for abuse and inappropriate behavior, and offer and award incentives to students for participation.

Anteater Groups shall allow students to report any abusive or inappropriate behavior, which is sent to UCI Administration.

Anteater Groups shall allow users to open a ticket for any troubleshooting or software-related issue to UCI OIT.

## **User Characteristics**

All users will have technical literacy and English literacy. UCI undergraduate and graduate students will be the primary users of Anteater Groups as the systems’ main goals focus on students’ interests, students’ mental health, and improving student connectivity. Students will have computer experience, and have knowledge of mobile application software systems. UCI Administration and CSWHP are the administrators of Anteater Groups, with UCI Administration taking responsibility of managing application use and overlooking development, and CSWHP being auxiliary users contributing knowledge on how to better student personal growth and provide wellness resources. Both UCI Administration and CSWHP will have knowledge on web-based subsystems. Group admins will have adequate experience with student and event management. Any user may have disabilities affecting their capability to navigate web-based and mobile application systems, so Anteater Groups shall accommodate through accessibility features not limited to changes in font and colors.

## **General Constraints**

1. The application shall only be available on mobile devices (iOS and Android) for undergraduate and graduate students.
2. The application shall have a web-based subsystem not limited to Google Chrome, Mozilla Firefox, Internet Explorer, Microsoft Edge, and Safari browsers, for wellness center staff and campus administration to view analytics and give administrative access to regulate and control the community and to reward students.
3. For optimal performance and experience, Anteater Groups require constant internet connection, whether using WiFi or cellular data. Without internet connectivity, users shall not be able to view any information on the application, especially not any up-to-date data.

## **Assumptions and Dependencies**

1. The login system shall be in line with the UCI system, and must be integrated with UCInetID and Duo Mobile.
2. Anteater Groups is expected to be accessible to all UCI students and administration and accommodate any disability-related needs which impact a user’s interaction with the system.
3. UCI alumni will not have access to Anteater Groups, as it is designed for current UCI students only.
4. Every group on the application must designate a group administrator.
5. Anteater Groups’ system and servers shall be able to support daily usage by the given population of UCI faculty and students.

## **Apportioning of Requirements**

These features shall be added in the future versions of the system:

1. Support facial recognition for logging on the app.
   1. Options to turn on or off facial recognition upon opening the app.
   2. Use facial recognition data features of the device.
2. Support for languages other than English (Spanish, Chinese, Korean, Vietnamese, Tagalog, etc.).
   1. Options to change the native language to different languages.
   2. Support English, Spanish, Chinese, Korean, Vietnamese, and Tagalog languages.
3. Provide 1-on-1 online mental health counselings through the app.
   1. Students are able to set up an appointment with a mental health specialist through the app.
4. Support off-campus or local community events.
   1. Allow students or event facilitators to add/create off-campus events.
   2. Allow the local community to add local events on the app under off-campus events.

# Specific Requirements

## **Essential Requirements**

### **Functional Requirements**

**ID: LOGIN**

Title: Login

Description: The system shall allow current students and onboarded group admins to login to Anteater Groups through their UCInetID and password login. Verifying login information through Duo is required when logging in with UCInetID information. The system shall allow users to login as “Guest” without the need to login with UCInetID information, and have “view-only” privileges of groups and group events.

Priority: High

Stability: High

Event/Use Case: Login, Register

Source: Goal Model 3: Login with UCI Login

Rationale: Alumni are unable to use Anteater Groups unless they are on boarded as group admins, using UCInetID login information will allow the school to check if a user is a current student.

Dependencies: Registered students must be current students and group admins must have a UCInetID.

**ID: CRTGRP**

Title: Creating Group

Description: The system shall allow UCI Students, Already Registered UCI Clubs/Organizations, and Event Facilitators such as counselors to create groups on the Anteater Groups app. The capacity limit is set by the group admin at the beginning of the group formation stage. Duplicate groups get notified when found.

Priority: High

Stability: High

Event/Use Case: A.1.3.16 CreateGroup

Source: Field Notes 5, 6, 10, 13, 39, 48, 63, 65, A.1.4.11 Domain model Group class

Rationale: For students to be able to join groups, there must first be a feature to create groups.

Dependencies: There needs to be a group admin to create groups.

**ID: SRCHGRP**

Title: Searching For Group

Description: The system shall allow students to search for groups available that are related to their academic and non-academic interests.

Priority: High

Stability: High

Event/Use Case: A.1.3.2 SearchGroup

Source: Field notes 4, 45

Rationale: Students need to be able to search for groups in order to connect with other students that have similar interests.

Dependencies: The user must be a student.

**ID: JOINGRP**

Title: Joining Group

Description: The system shall allow students to join groups that they wish to be a part of.

Priority: High

Stability: High

Event/Use Case: A.1.3.3 JoinGroup

Source: Field notes 6

Rationale: Students should be able to join groups that align with their interests so that they can interact with students who have the same interests. Joining a group would allow them to see events related to the group.

Dependencies: The group must be created and have an onboarded admin(s) before it can be joined by students.

**ID: EDTPRF**

Title: Edit User Profile

Description: The system shall allow students to edit their profile by adding a profile picture, short bio description, contact info, and social media accounts.

Priority: Medium

Stability: Medium

Event/Use Case: A.1.3.1 EditProfile

Source: Field notes 37, 44, 66b

Rationale: Students should be able to edit their profile with information they want to make public to connect with students that may share the same interests.

Dependencies: The user must be a current UCI student.

**ID: MNGFRND**

Title: Adding and Removing User as a Friend

Description: The system shall allow students to manage their friends list by adding other students as friends or removing students from their friends list. Students shall be able to view a list of friends on their profile. Students will be able to privately message students that are on their friends list.

Priority: Medium

Stability: Medium

Event/Use Case: A.1.3.4 SendFriendRequest, A.1.3.8 RemoveFriend

Source: Field notes 27, 49, 67, 70, 79

Rationale: Students will feel more connected if they have a social network of friends that they can communicate and plan social gatherings with.

Dependencies: None

**ID: RPTUSER**

Title: Report User For Inappropriate Behavior

Description: The system shall allow students to 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.

Priority: High

Stability: Medium

Event/Use Case: A.1.3.6 ReportBehavior

Source: Field Notes 12, 25, 29, 52, 69, A.1.4.6 Domain model Report class

Rationale: The user will have conducted an inappropriate behavior according to the AnteaterGroup’s policy for that user to get reported and punished accordingly.

Dependencies: The user must be a mutual member of a group or a friend of the reporting student.

**ID: MNEVNT**

Title: Manage Event

Description: The system shall allow group admins to create and broadcast an event within their group that is either private to current members or public to all Anteater Groups users. Event information shall include a name, date/time, event category (ex: protest, entertainment), and guidelines regarding capacity limits or Covid regulations. The system shall allow students to RSVP these events, pushing them onto their device’s calendars, and generate a QR code for group admins to present at events for students to scan their attendance in.

Priority: High

Stability: High

Event/Use Case: A.1.3.10 BroadcastEvent, A.1.3.11 RSVPEvent, A.1.3.12 ScanAttendanceQR

Source: Field notes 5, 6, 82, Goal model “Improve students’ social connectedness,” A.1.4.12 Domain model Event class

Rationale: Only group admins can promote events to reduce potential noise if students were able to broadcast them. Restricting this to group admins will ensure that event information is precise. Event information should include a type as protest events may require UCI Police as security.

Dependencies: Only registered users can RSVP and scan the QR codes for events. Only group admins can create and broadcast events.

**ID: METRICS**

Title: View Usage Metrics

Description: The system shall allow UCI Administration the ability to view metrics on a single web-page. They shall be able to view the number of students registered, number of logins, group participation, group headcount.

Priority: High

Stability: High

Event/Use Case: A.1.3.5 ViewUsageMetrics

Source: Field notes 38, 50, case study,

Rationale: UCI Administration should be able to view usage metrics on the application to take any appropriate action to promote engagement among students and their groups.

Dependencies: The user must be a UCI Administration.

**ID: CAMPRES**

Title: Utilize Campus Resources

Description: The system shall allow students to utilize campus resources that are not run by Anteater Groups such as UCI Health Safety Escorts, UCI Library study rooms, and UCI Wellness mental health specialists. They are able to access these resources within Anteater Groups, either through provided contact information, library API integration, and through the ability to make appointments with Wellness Center specialists.

Priority: Low

Stability: Medium

Event/Use Case: A.1.3.13 UtilizeCampusResource

Source: Field notes 26, 52, 61 and Goal model “Improve student involvement with campus resources and organizations,” A.1.4.10 Domain model Campus Resource class

Rationale: The system should provide contact information of UCI Health Safety Escorts as students may feel unsafe traveling to and from events. Students should be able to book Library study rooms through the app as well if they want to meet with their group members at the library. As the app is also managed by CSWHP, the system should allow students to make appointments with health specialists through the app.

Dependencies: None.

**ID: TICKET**

Title: Raise Ticket

Description: The system shall allow any user to raise a ticket to bring up any technical issues or concerns they encounter when using the application.

Priority: High

Stability: High

Event/Use Case: None

Source: Field notes 92, A.1.4.5 Domain model Ticket class

Rationale: Issues and concerns regarding the system should be reported so that users will not face any issues during their use.

Dependencies: A support team must be available to address the issues and concerns as they arise.

**ID: GRPREV**

Title: Write Group Review

Description: The system shall allow students to contribute to the evaluation of groups by writing group reviews. Group reviews are ranked by the 5-star-scale where 1 star being the worst and 5 stars being the best.

Priority: Medium

Stability: Low

Event/Use Case: None

Source: Fields note 71, 72

Rationale: Students should be able to write a review for any groups when joined.

Dependencies: None.

**ID: CHAT**

Title: Chat Feature

Description: The system shall allow students to chat through message posts to one another within the groups they are members of and may include file attachments and images. The system shall also allow students to chat privately in subgroups and give them the option to video and voice chat within these subgroups.

Priority: High

Stability: Medium

Event/Use Case: A.1.3.9 ChatWithOthers, A.1.4.7 Domain model Chat class

Source: Field Notes 33, 91

Rationale: Students should be able to chat to interact with other students, and engage with one another.

Dependencies: None.

### **Non-functional Requirements**

**ID: SECUR**

Title: Security

Description: The system shall securely transfer and store data using modern encryption standards and protocols. The system shall use HTTPS, AES-256 encryption, and random delays. Users must have an account with the system and authenticate through Duo Mobile. When a student graduates or a user is no longer part of the system, their account will be terminated and all associated data will be deleted.

Priority: High

Stability: High

Event/Use Case: None

Source: Field notes 30, 104

Rationale: The information must be secure to protect sensitive information and prevent bots from any malicious attacks.

Dependencies: Users must be logged into the system.

**ID: SAFE**

Title: Safety

Description: The system shall ensure that students are safe and protected from any threat. Students shall report students that exhibit any inappropriate behavior, and UCI administration will be able to review the report to take the appropriate action. A machine learning algorithm will be implemented to also detect any behavior that is deemed inappropriate. The system shall also list resources that ensure safety, such as UCI Health Safety Escorts.

Priority: High

Stability: High

Event/Use Case: None

Source: Field notes 25, 29

Rationale: Students should feel comfortable and safe while using the system so they continue to engage with the application and other students.

Dependencies: None

**ID: PERF**

Title: Performance

Description: The system shall support at least 10,000 users with a latency under 1 second.

Priority: High

Stability: High

Event/Use Case: None

Source: Field notes 16

Rationale: The system shall be able to support actions from all students using the application so they are able to get the most up-to-date information.

Dependencies: The user must be connected to the internet.

**ID: ACCESS**

Title: Accessibility

Description: The system shall comply with the Americans with Disabilities Act (ADA). The system shall incorporate assistive technologies such as varying font and color for the visually impaired. The system shall incorporate a screen reader and captions for any embedded videos.

Priority: High

Stability: Medium

Event/Use Case: None

Source: Field notes 54

Rationale: Anteater Groups should not discriminate towards any student in a way that will prevent them from connecting with other students and exploring their interests through the app based on a disability.

Dependencies: None

**ID: AVAIL**

Title: Availability

Description: The system shall be online 24/7 excluding scheduled maintenance times and unexpected crashes that require immediate maintenance.

Priority: High

Stability: Medium

Event/Use Case: None

Source: Field notes 40

Rationale: The system needs to be online in order for students to interact with one another. The system should not be limited to certain hours of the day as students have varying sleep and work schedules that should not conflict with their ability to freely use Anteater Groups.

Dependencies: There should be available developers on-call or a technical support team to handle ticket requests that may occur at any time of day.

### **External Interface Requirements**

#### User Interfaces

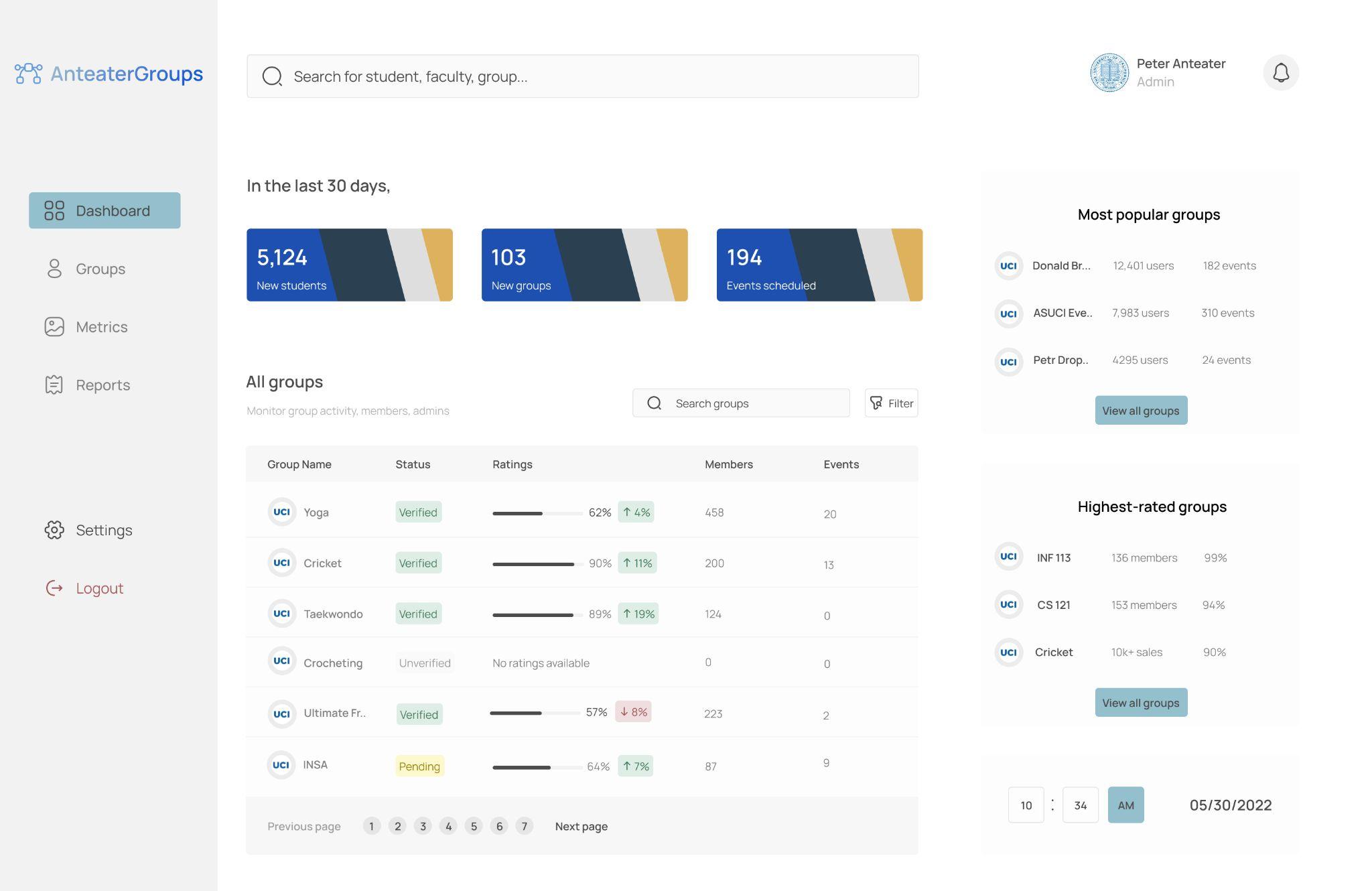


Figure 1: Dashboard of Anteater Groups’ web-based subsystem for UCI administrators to access all students, groups, and metrics.

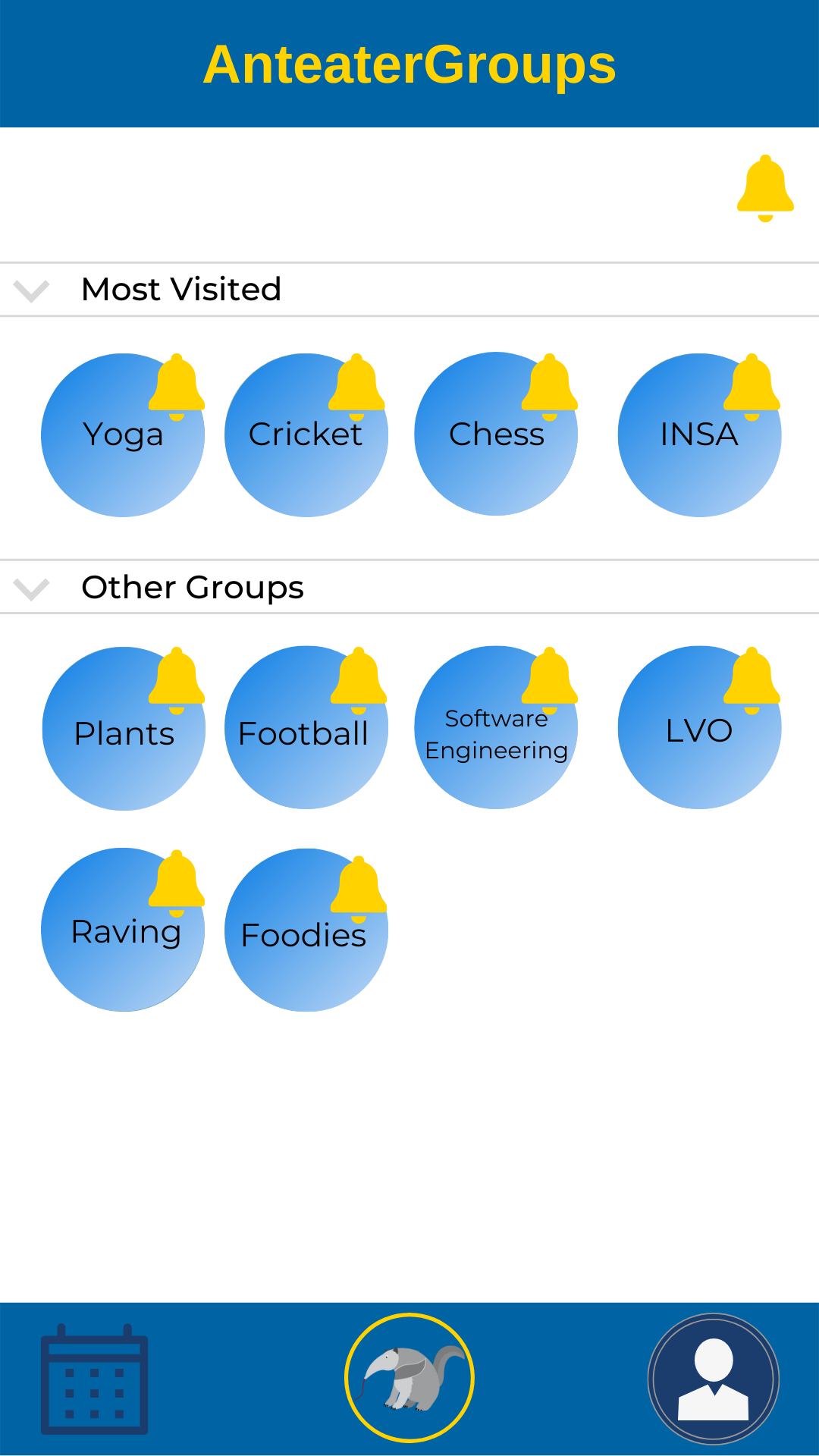


Figure 2: Homepage for Anteater Groups application for students featuring a list of all joined groups.

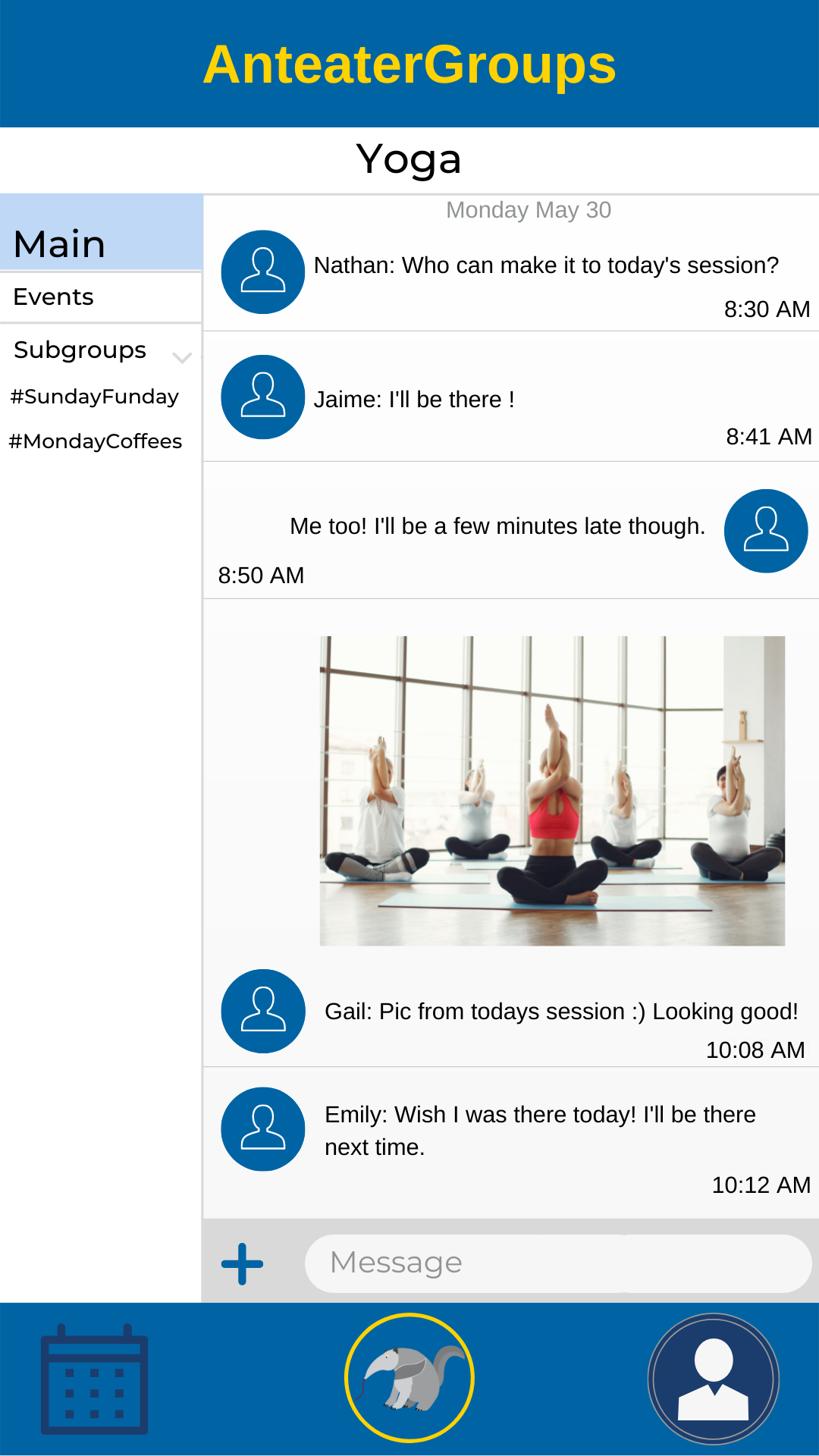


Figure 3: View of a group’s main chat through a group member’s perspective.

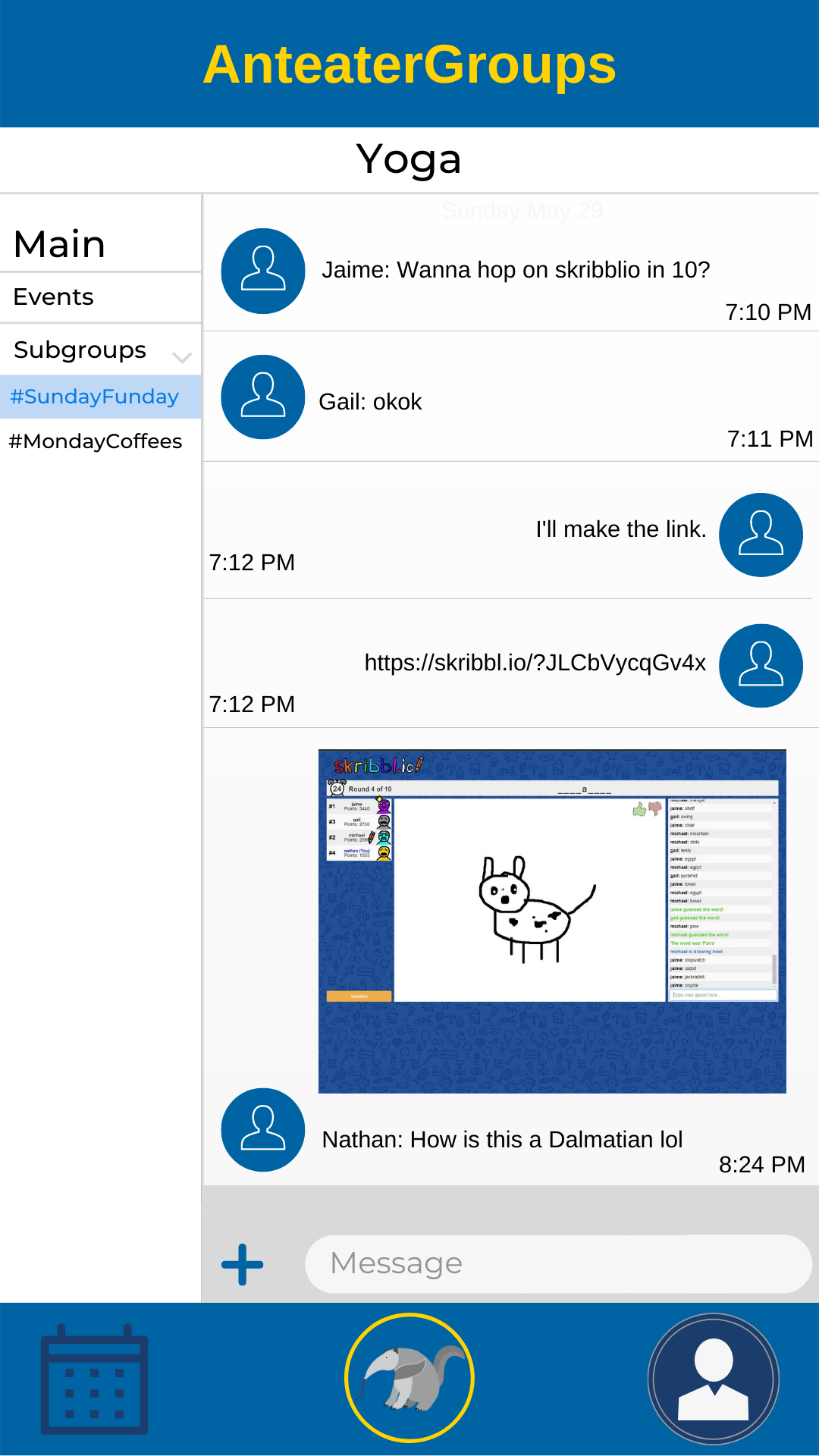


Figure 4: View of a subgroup (SundayFunday) within a main group (Yoga).

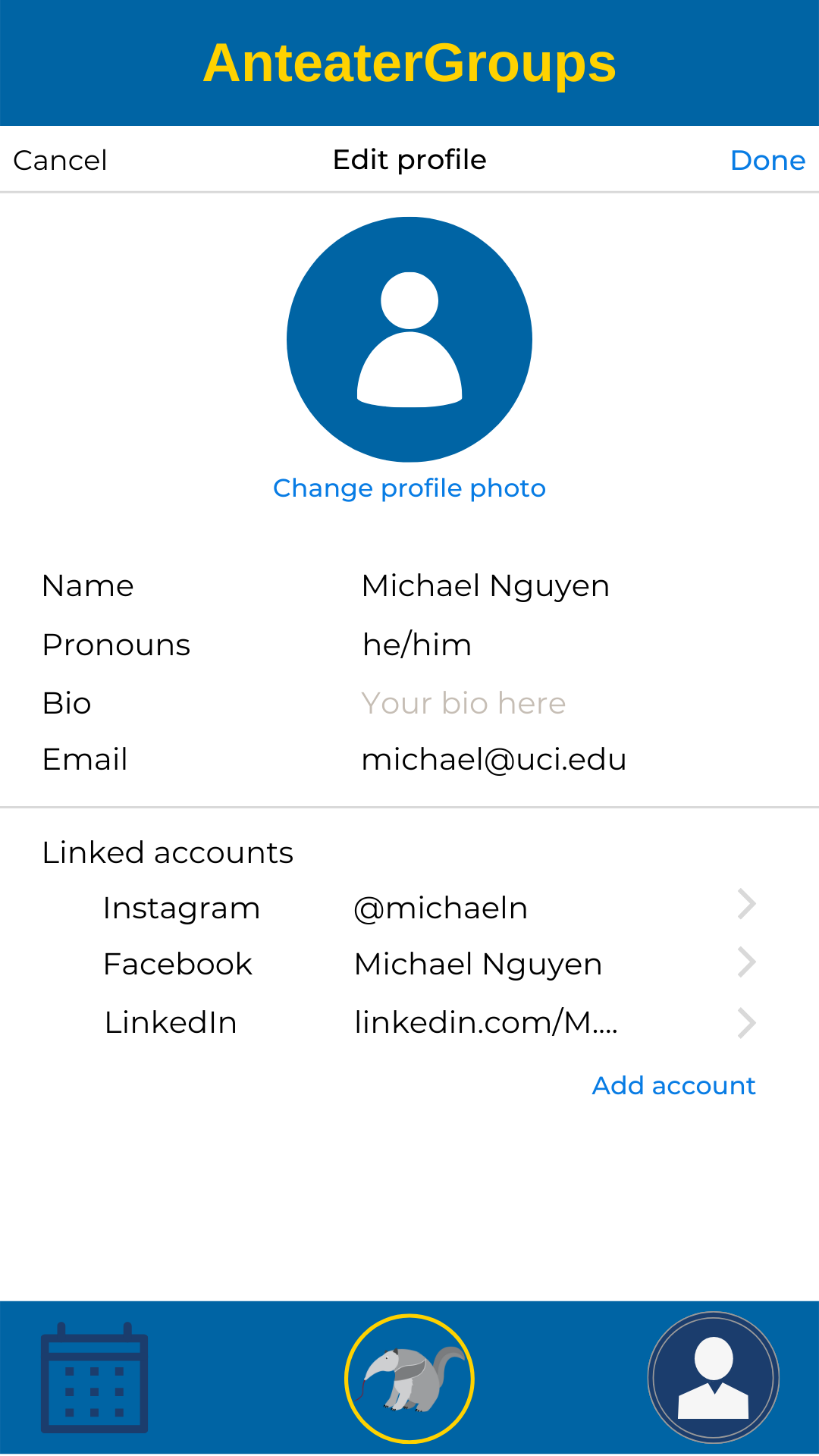


Figure 5: Display of user’s edit profile page, where they can customize their profile.

#### Hardware Interfaces

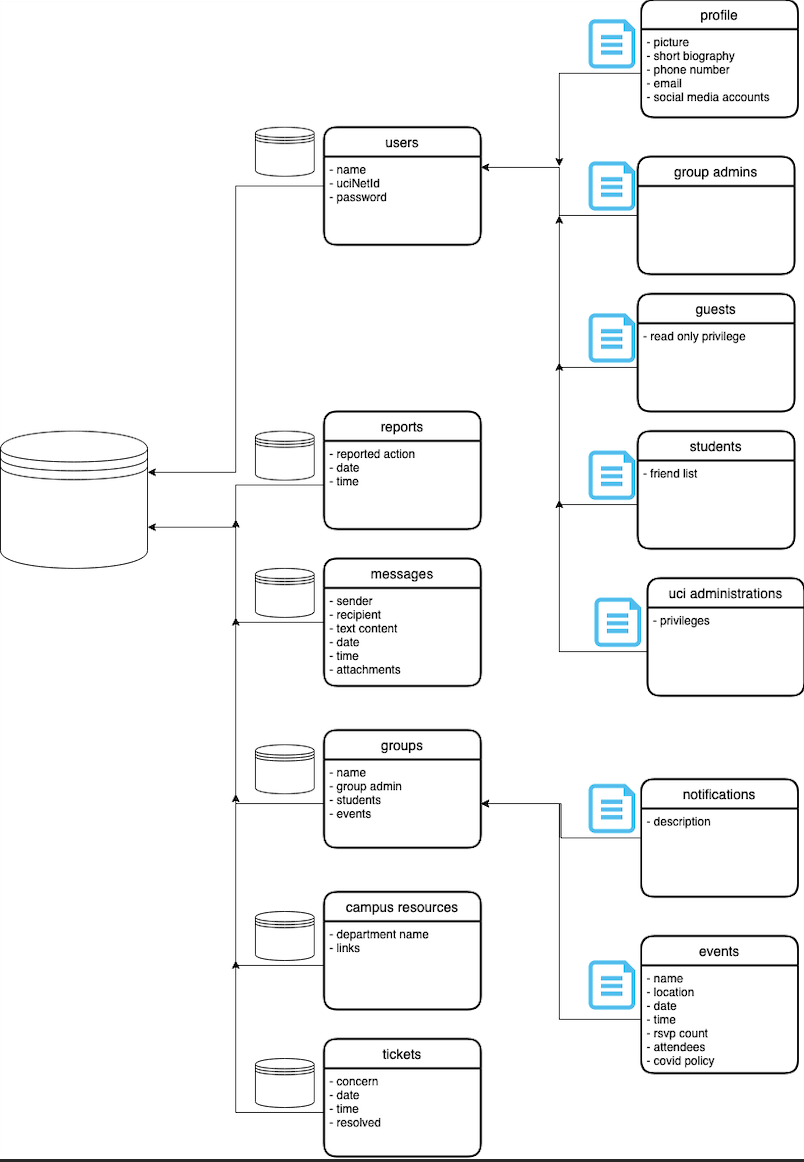
1. Web Browser
   1. Chrome version 5.0.375 or newer with layout engine WebKit 533 or better, Internet Explorer version 11.0.220 with support for operating systems Windows 7 or later and Server 2008 R2 or later, and Firefox Mozilla version 91.9.1 or later for Windows and macOS with minimum OS version required Windows 7, Server 2008 R2 or later and macOS 11 or later. Minimum system required for all OS platforms is A 64-bit x86 processor, 4GB of RAM, 16 of internal storage, and USB booting support and full administrator access to the BIOS.
2. Android Mobile Device
   1. Android version 9.0 (Pie) and newer, minimum RAM of 3 GB, Internal Storage of 32GB or more, Qualcomm Snapdragon (version 630 and higher), Samsung Exynos, Hisilicon Kirim or any other 2.5 Ghz quad-core application processors.
3. iOS Mobile Device
   1. The iOS versions of 10.0.0 or newer are required to run on the iOS supported hardware. The minimum hardware system requirements are Apple A6 and A6X processors with x32-64 and x64 architectures, or newer version processors.

#### **Software Interfaces**

The internal and external softwares that interact with the Anteater Groups:

| **Software** | **Description** |
| --- | --- |
| Web Browsers  (Google Chrome, Safari, Firefox) | Anteater Groups will use the Chrome browser, Safari browser, and/or Firefox browser to interact with web-based users. |
| Mobile Devices  (iOS, Android, Windows, macOS) | Anteater Groups will use iOS, Android, Windows, and/or macOS operating systems to run its application for mobile devices. |
| WebAuth \* | WebAuth works in conjunction with web browsers (Chrome, Safari, Firefox) to allow students and admins to sign in onto Anteater Groups web-app. |
| Duo Mobile \* | Duo Mobile works in conjunction with mobile devices using iOS, Android, Windows, and macOS as the operating system to allow students and admins to sign onto the Anteater Groups mobile app. |
| Postgresql \* | Anteater Groups will use SQL to store user data and other app data using Postgresql database technology. |

### **Logical Data Model**



The database shall contains the following entities:

**Users:** name, uciNetID, password. There are four types of user: students, group admins, guests, and uci administrators.

1. Students will have a student name, student uciNetID, password, picture, short description, phone number, email, social media links, and friends list.
2. Group admins will have everything that is in the users class.
3. Guests will only have read-only privilege.
4. UCI admins will have everything in the users class and some more admin privileges.

**Reports:** report action, date, time.

**Messages:** sender, recipient, text content, date, time, attachments.

**Groups:** name, group admin, students, events. There are two subcategories in groups: notifications and events.

1. Notifications: description.
2. Events: name, location, date, time, RSVP count, attendees, covid policy.

**Campus resources:** department name and links.

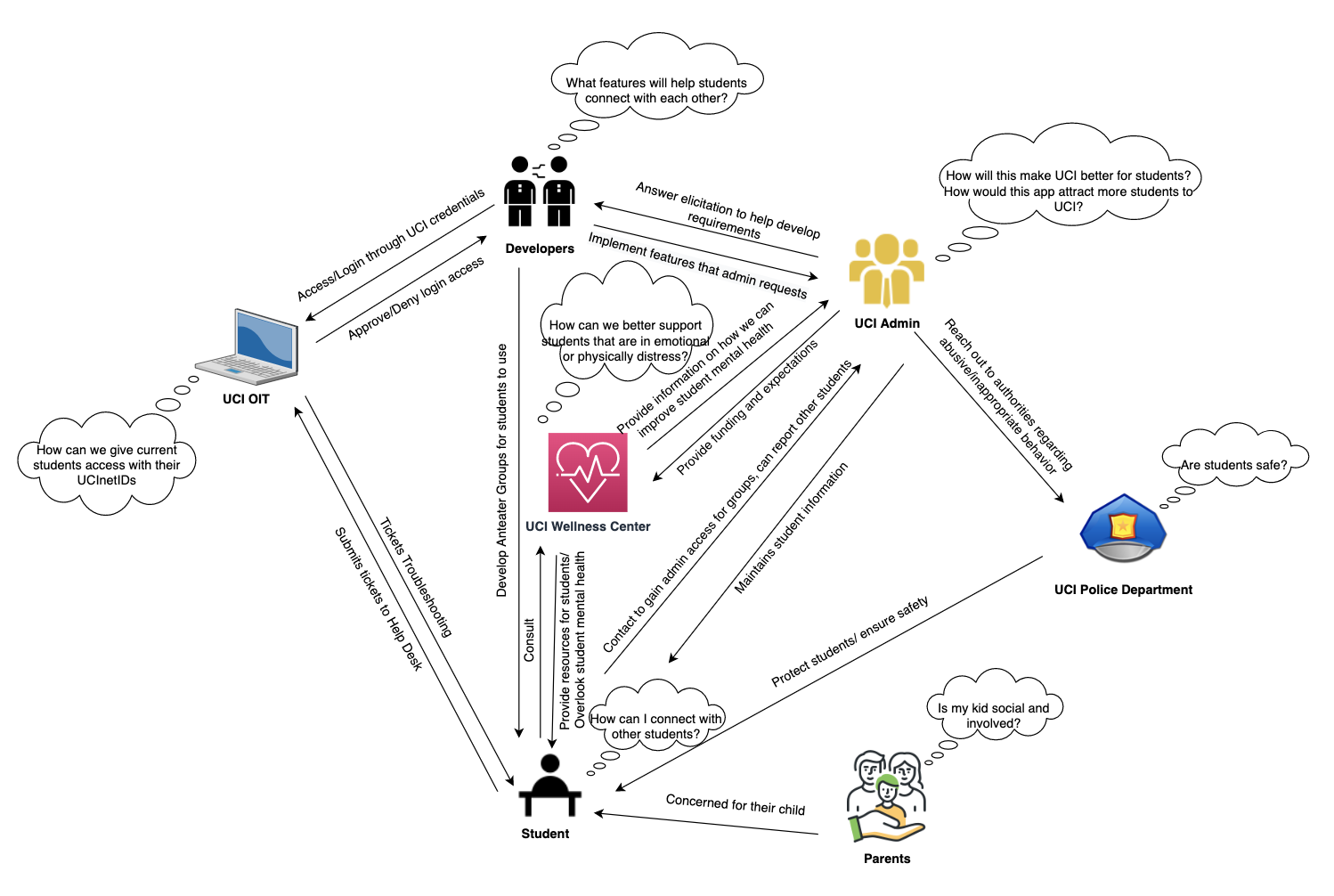
**Tickets**: concern, date, time, resolved status.

# Appendix

**A.1 Analysis Models:**

## **A.1.1 Stakeholder Model**

Rich Picture

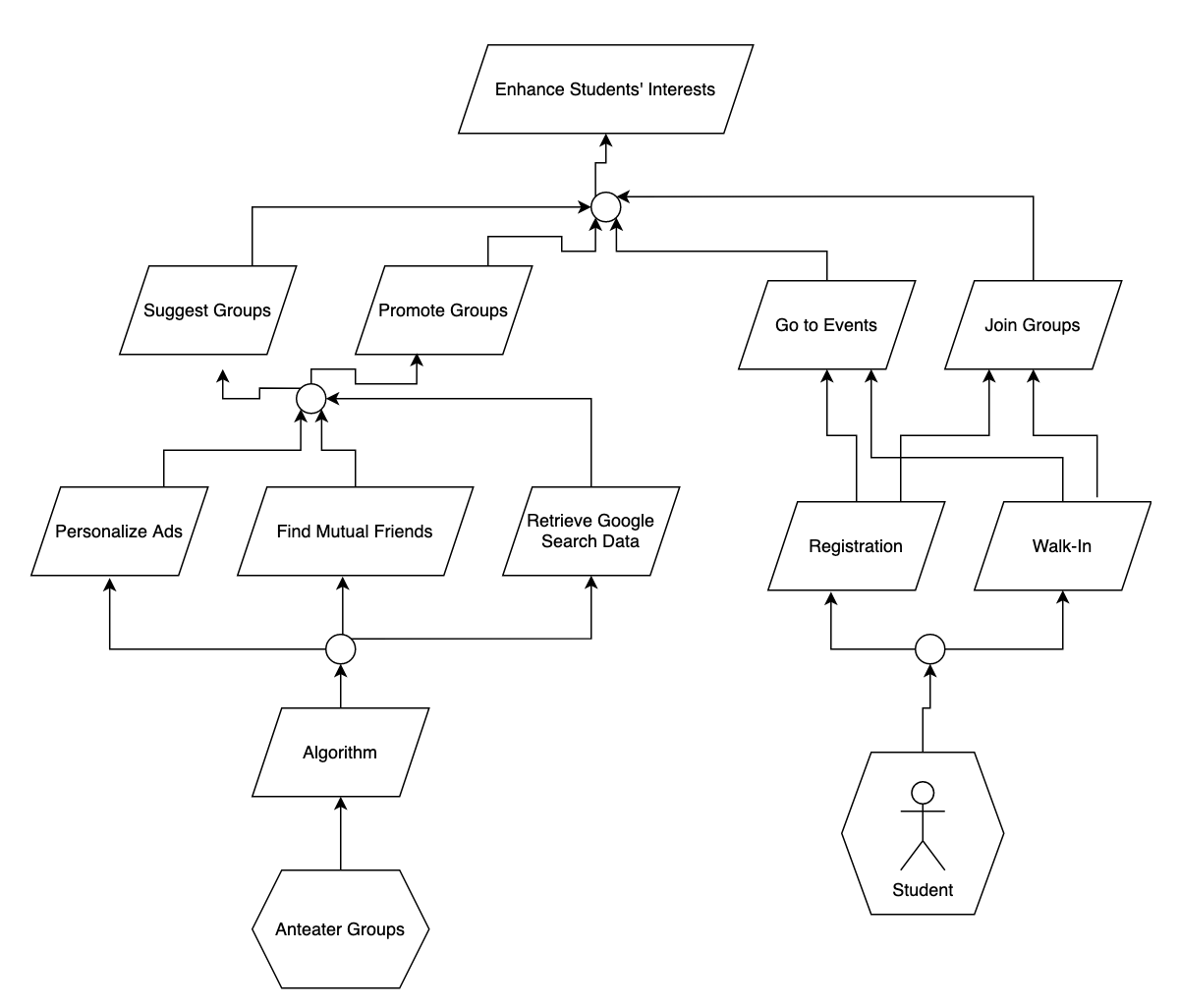


Textual Stakeholder Analysis

* UCI Administration
  + Priority: High
  + Relation: The facilitator and approval committee overlooking the overall Anteater Groups project. Any expectations, requirements, and specifications are outlined by the UCI Administration. They are in partnership with the UCI Center for Student Wellness and Health Promotion and developers to create Anteater Groups.
  + Expertise: UCI Administration has full knowledge of the different groups on campus, including but not limited to schools (ex. Henry Samueli School of Engineering), clubs, support services, commissions, centers, offices (ex. Office of the Vice Provost for Research), and departments.
  + Primary Concern(s): Administration wants the app to be successful among students as they build a stronger community, and possibly improve the outlook of UCI to the rest of the world.
* UCI Center for Student Wellness and Health Promotion
  + Priority: High
  + Relation: The UCI Center for Student Wellness and Health Promotion is overlooking student mental health. They overlook student interaction and provide resources for students regarding wellness and strive for students to connect with one another. They want to work together with UCI Administration and developers to create Anteater Groups.
  + Expertise: The UCI Center for Student Wellness and Health Promotion are experts at mental health understanding. They understand it is difficult to connect with other students and make friends.
  + Primary Concern(s): They want students to improve their mental health through enhanced social connectedness and personal growth. They want students to learn and grow in their personal interests and overall engage more with their community.
* UCI OIT
  + Priority: Medium
  + Relation: UCI OIT manages students’ UCInetIDs that provide students with access to UCI online services. Students shall use these UCInetIDs to log into Anteater Groups. OIT is responsible for giving proper access to current UCI students and working together with developers to allow students to login.
  + Expertise: They are experts at information security and management. They know how to work with other team developers to integrate UCInetID logins onto other applications.
  + Primary Concern(s): They are concerned about properly providing access to Anteater Groups and associating students with their login information. Only current students and admins can use the app currently.
* UCI Students
  + Priority: High
  + Relation: The sole focus of the application is the students. Their needs and expectations are carefully considered in the designing and drafting of the Anteater Groups app. They can directly reach out to UCI Admin to report student behavior or ask for admin permissions, and can depend on the UCI Center for Student Wellness and Health Promotion to overlook their mental health.
  + Expertise: Beginner; Learning how to use the app and navigating through the app to grow their interests and find their relatable groups to join.
  + Primary Concern(s): Anteater Groups can address their issues or not.
* UCI Parents
  + Priority: Low
  + Relation: Parents and guardians of UCI students. Parents will want to make sure that their child is able to connect with others and join social groups that promote connectivity and help them to find their interests.
  + Expertise: Parents are knowledgeable about the wellbeing of their child and expect their school to properly take care of their children.
  + Primary Concern(s): Parents want to know how their child is doing in school.
* UCI Police Department
  + Priority: Medium
  + Relation: UCIPD addresses and investigates reported cases provided by UCI Administration on Anteater Groups if they require attention from law enforcement. They are the last in the chain to address student-reported issues on the application.
  + Expertise: The police department are experts in handling situations that deal with rules or the law. They are able to safely navigate dangerous situations and make sure that people are protected.
  + Primary Concern(s): The police would be concerned with the safety and well-being of students and administration. They would want to make sure that students who are not behaving appropriately will be monitored and have consequences for their actions.
* Developers
  + Priority: Medium
  + Relation: The software developers program all aspects of the application as well as address and fix any bugs found by students. They work together with the UCI Center for Student Wellness and Health Promotion and UCI Administration to develop the app that they want. The UCI Center for Student Wellness and Health Promotion and UCI Admins instruct the developers on how student interaction can be improved.
  + Expertise: The developers have the most knowledge on how the algorithm of Anteater Groups works, as well as the best frameworks and technologies to develop the app for optimal performance and meet specifications.
  + Primary Concern(s): They are concerned with implementing the right features that will satisfy the goals of Anteater Groups. They want to accurately document their customer’s needs, implement them, and provide a positive user experience for students and admins.

## **A.1.2 Goal-Oriented Analysis Models**

Model 1: “Enhance Students’ Interests”



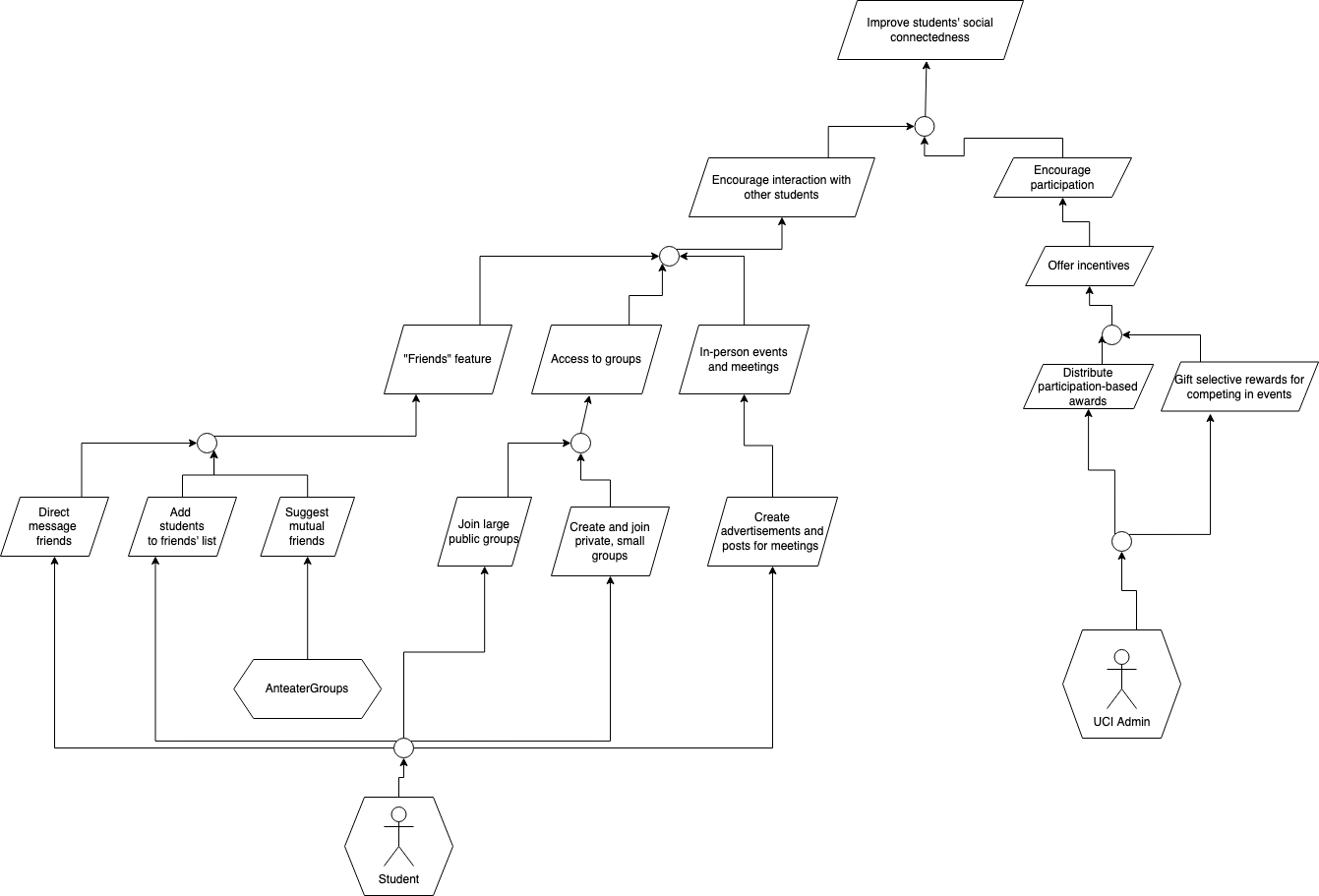
Goal priorities are categorized as Low, Medium, High.

Source numbers refer to the numbered list in the Field Notes section from elicitation sessions.

1. **Enhance students’ interests**
   1. Definition: Anteater Groups shall enhance student’s current interests through using the app.
   2. Type: Soft
   3. Source: Case StudyPriority: High
2. **Suggest Groups**
   1. Definition: Students shall see suggestions for groups on a floating banner or a pop-up as personalized ads based on their online searches and browsing history on the app.
   2. Type: Achieve
   3. Source: 22, 27, 67
   4. Priority: Medium
3. **Promote Groups**
   1. Definition: Students shall see promotion based on their interests and connections with other students based on their mutual friend list.
   2. Type: Achieve
   3. Source: 22, 27, 67
   4. Priority: High
4. **Join Groups**
   1. Definition: Students shall be able to register to join groups.
   2. Type: Achieve
   3. Source: 6
   4. Priority: High
5. **Go to Events**
   1. Definition: Students shall be able to go directly to events as walk-in.
   2. Type: Achieve
   3. Source: 6
   4. Priority: Medium
6. **Personalize Ads**
   1. Definition: Students shall see personalized ads to join groups based on their interests.
   2. Type: Achieve
   3. Source: 67
   4. Priority: High
7. **Find Mutual Friends**
   1. Definition: Students shall see mutual friends and for connections through friend requests.
   2. Type: Achieve
   3. Source: 27, 70
   4. Priority: High
8. **Retrieve Google Search Data**
   1. Definition: Students shall see suggested groups based on their browsing history from Google.
   2. Type: Achieve
   3. Source: 67
   4. Priority: Medium
9. **Registration** 
   1. Definition: Students shall be able to register to join groups.
   2. Type: Achieve
   3. Source: 45
   4. Priority: High
10. **Walk-in**
    1. Definition: Students shall have the option to join groups as walk-in.
    2. Type: Achieve
    3. Source: 45
    4. Priority: Medium
11. **Algorithm**
    1. Definition: System shall use machine learning algorithms to implement personalized recommendations.
    2. Type: Maintain
    3. Source: 25, 67
    4. Priority: High

# 

Model 2: “Improve students’ social connectedness”

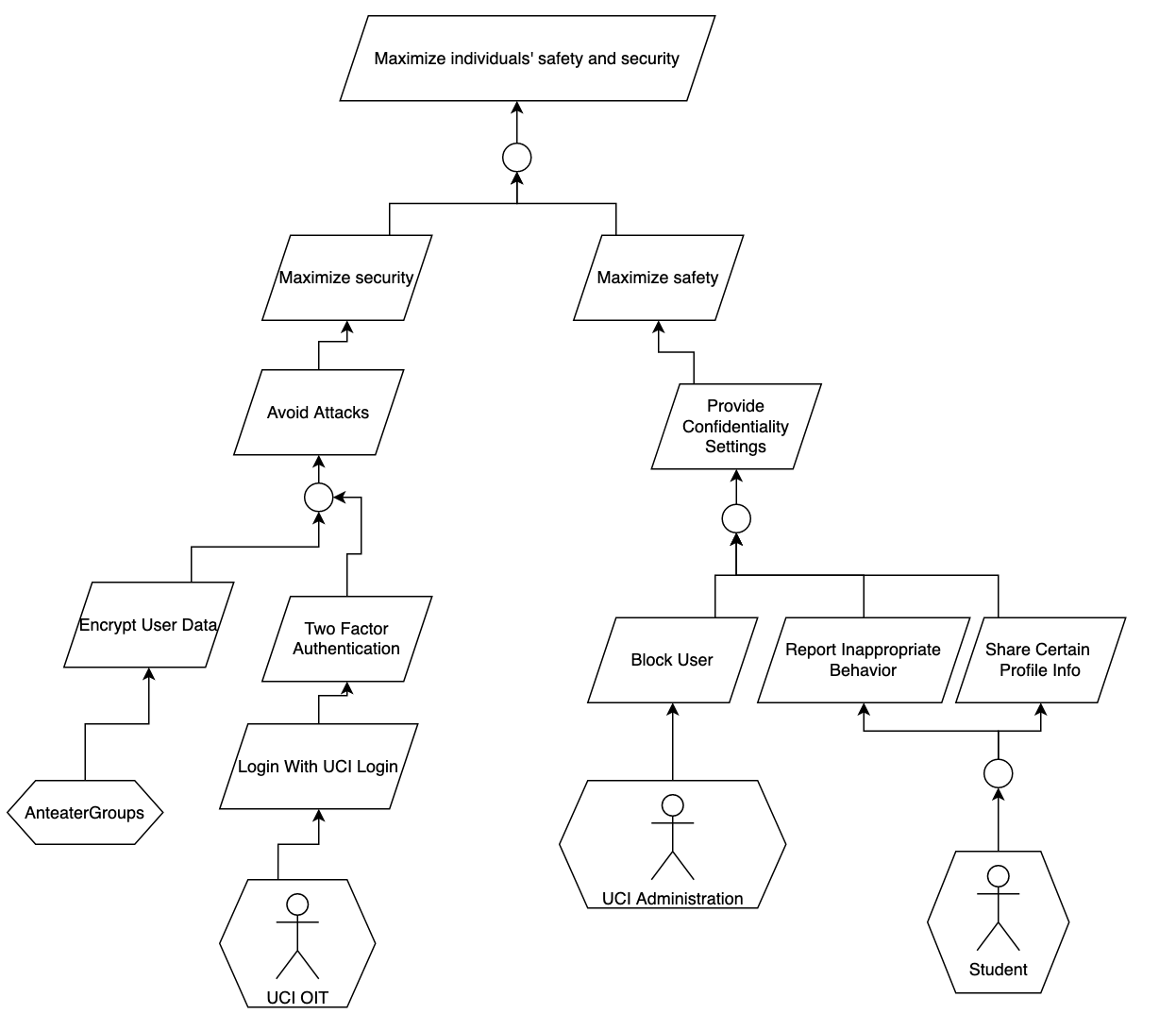


Goal priorities are categorized as Low, Medium, High.

Source numbers refer to the numbered list in the Field Notes section from elicitation sessions.

1. **Improve students’ social connectedness**
   1. Definition: Anteater Groups shall improve students’ social connectedness through their use of the app.
   2. Type: Soft
   3. Source: Case Study
   4. Priority: High
2. **Encourage interaction with other students**
   1. Definition: The system should promote communication between UCI students.
   2. Type: Soft
   3. Source: 2, 5, 6, 11, 19, 27, 30
   4. Priority: High
3. **Encourage participation**
   1. Definition: The system should promote students’ participation in events.
   2. Type: Soft
   3. Source: 7, 9, 42
   4. Priority: High
4. **Offer incentives**
   1. Definition: UCI Administration shall offer incentives to students to encourage them to participate in activities.
   2. Type: Achieve
   3. Source: 7, 9, 42
   4. Priority: High
5. **“Friends” feature**
   1. Definition: Anteater Groups shall give students the ability to have a list of friends.
   2. Type: Maintain
   3. Source: 49
   4. Priority: High
6. **Access to groups**
   1. Definition: Students shall have access to large public groups and smaller private groups.
   2. Type: Maintain
   3. Source: 63, 64
   4. Priority: High
7. **Establish in-person events and meetings**
   1. Definition: Students shall be able to attend events and meetings in-person on campus.
   2. Type: Achieve
   3. Source: 46, 47
   4. Priority: High
8. **Distribute participation-based rewards**
   1. Definition: Students shall receive rewards for participating in events.
   2. Type: Achieve
   3. Source: 9, 42
   4. Priority: High
9. **Gift selective rewards for competing in events**
   1. Definition: Administration shall gift select students rewards for winning raffles or in-person events.
   2. Type: Achieve
   3. Source: 7, 9, 42
   4. Priority: Medium
10. **Direct message friends**
    1. Definition: Students shall always be able to direct messaging those on their friends list.
    2. Type: Maintain
    3. Source: 33
    4. Priority: High
11. **Add students to friends’ list**
    1. Definition: Students shall always be able to add their peers to their friends list.
    2. Type: Maintain
    3. Source: 49, 70
    4. Priority: High
12. **Suggest mutual friends** 
    1. Definition: Anteater Groups shall suggest befriending students who have mutual friends with the user.
    2. Type: Maintain
    3. Source: 27, 70
    4. Priority: Low

Model 3: “Maximize individuals' safety and security”

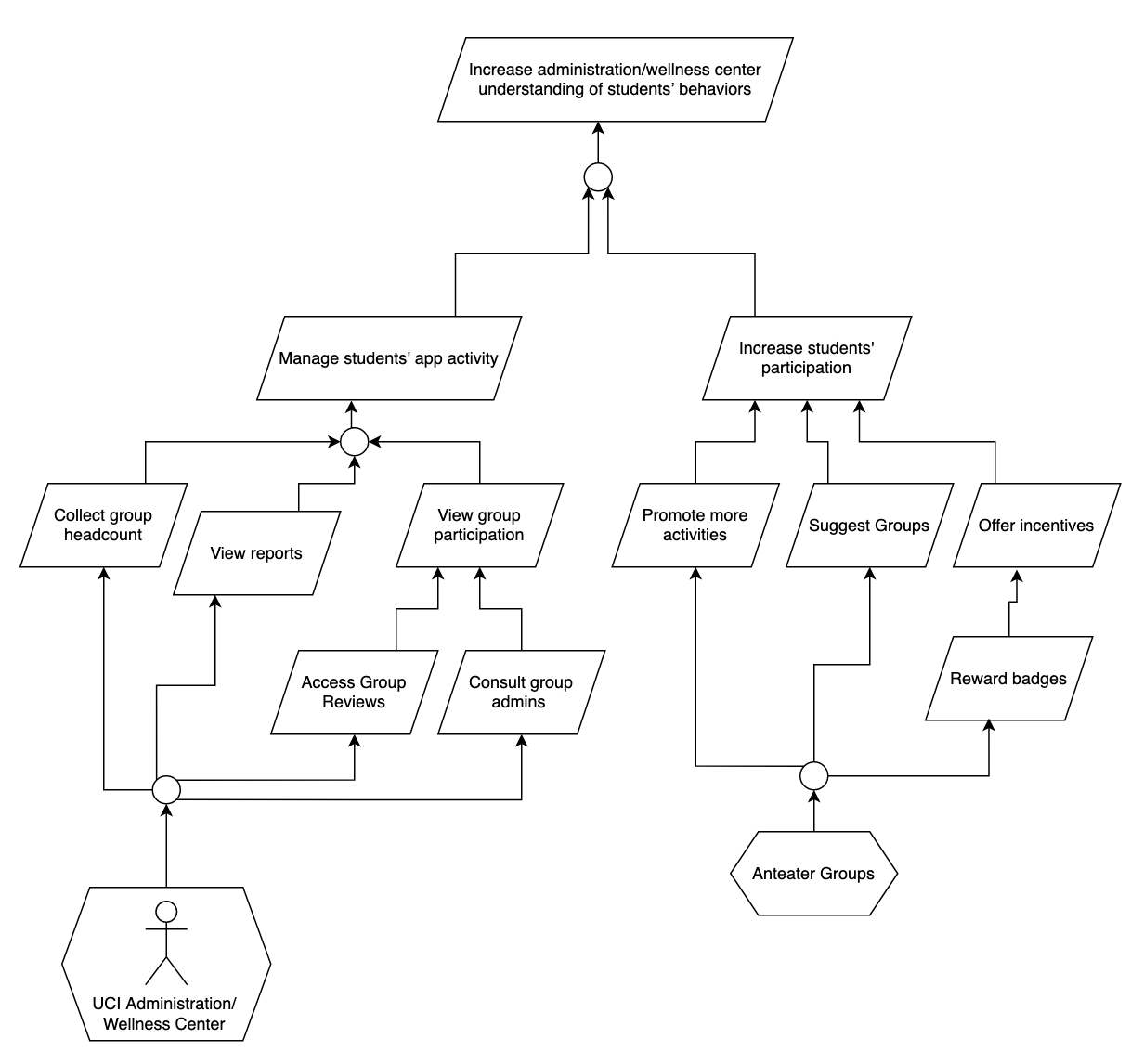


Goal priorities are categorized as Low, Medium, High.

Source numbers refer to the numbered list in the Field Notes section from elicitation sessions.

1. **Maximize individuals’ safety and security**
   1. Definition: Anteater Groups should maximize the safety and security of students while they are using the application.
   2. Type: Soft
   3. Source: 59, 30
   4. Priority: High
2. **Maximize security**
   1. Definition: The system shall maximize security so that student data is protected.
   2. Type: Soft
   3. Source: 59, 30
   4. Priority: High
3. **Avoid Attacks**
   1. Definition: The system should avoid attacks from students and other entities to keep all data secure.
   2. Type: Avoid
   3. Source: 59, 30
   4. Priority: High
4. **Encrypt User Data**
   1. Definition: User data must be encrypted so that sensitive information is protected.
   2. Type: Maintain
   3. Source: 30
   4. Priority: High
5. **Two Factor Authentication**
   1. Definition: The system shall have two factor authentication in order to verify that the user logging in is the actual user.
   2. Type: Maintain
   3. Source: 59
   4. Priority:
6. **Login with UCI Login**
   1. Definition: Users must log in with their UCI Login Credentials in order to use Anteater Groups.
   2. Type: Maintain
   3. Source: 59
   4. Priority: Medium
7. **Maximize safety**
   1. Definition: Anteater Groups must maximize safety so users feel safe and comfortable while using the application.
   2. Type: Soft
   3. Source: 25, 44
   4. Priority: High
8. **Provide Confidentiality Settings**
   1. Definition: There must be confidentiality settings so that users feel comfortable.
   2. Type: Maintain
   3. Source: 25, 44
   4. Priority: High
9. **Block User**
   1. Definition: UCI Administration shall be able to block users from using Anteater Groups if certain users display inappropriate behavior.
   2. Type: Achieve
   3. Source: 25
   4. Priority: High
10. **Report Inappropriate Behavior**
    1. Definition: Students shall be able to report other students that display inappropriate behavior.
    2. Type: Achieve
    3. Source: 25, 52
    4. Priority: High
11. **Share Certain Profile Info**
    1. Definition: Students should be able to share personal information that they want other users to see.
    2. Type: Achieve
    3. Source: 44
    4. Priority: Medium

Model 4: “Increase administration/wellness center understanding of students’ behaviors”



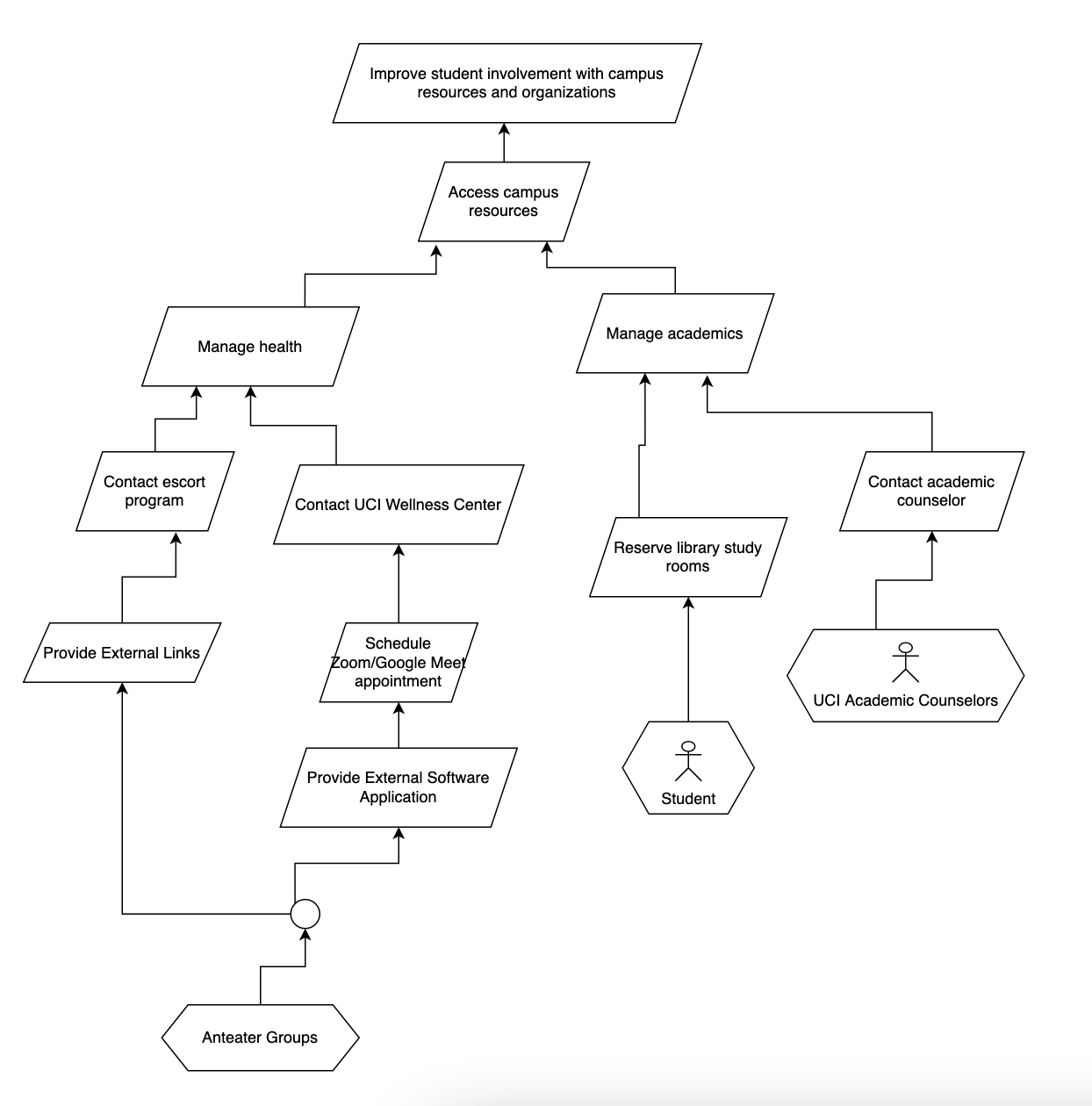
Goal priorities are categorized as Low, Medium, High.

Source numbers refer to the numbered list in the Field Notes section from elicitation sessions.

1. **Increase administration/wellness center understanding of students’ behaviors**
   1. Definition: UCI Administrators and the UCI Wellness Center shall be able to increase their understanding of students’ behaviors.
   2. Type: Soft
   3. Source: Case Study
   4. Priority: High
2. **Manage students’ app activity**
   1. Definition: UCI Administrators and the UCI Wellness Center shall be able to view usage statistics of students on a centralized dashboard.
   2. Type: Achieve
   3. Source: 38, 50
   4. Priority: High
3. **Increase students’ participation**
   1. Definition: Student activity on the app shall be maximized to improve quality of data retrieved on the app.
   2. Type: Soft
   3. Source: 7, 9, 42
   4. Priority: High
4. **Collect Group Headcount**
   1. Definition: UCI Administrators and the UCI Wellness Center shall be able to view the amount of students that have joined in each group.
   2. Type: Achieve
   3. Source: 38, 50
   4. Priority: Medium
5. **View reports**
   1. Definition: UCI Administrators and the UCI Wellness Center shall be able to view reports, along with the previous 4-5 chats where the user flagged as inappropriate in the case of chat reports, in order to address student behavior.
   2. Type: Achieve
   3. Source: 12, 25, 29, 52, 69
   4. Priority: Medium
6. **View group participation**
   1. Definition: UCI Administrators and the UCI Wellness Center shall see values on how many students have created groups and view data on group performance.
   2. Type: Achieve
   3. Source: 38, 50
   4. Priority: High
7. **Promote more activities**
   1. Definition: Admins shall have the authority to promote group activities to prevent unnecessary noise from students promoting activities.
   2. Type: Maintain
   3. Source: 5, 50
   4. Priority: Medium
8. **Suggest groups**
   1. Definition: Students shall be suggested groups based on their interests, mutual friends, and Google Search activity.
   2. Type: Achieve
   3. Source: 2, 67
   4. Priority: Medium
9. **Offer incentives**
   1. Definition: UCI Administrators and the UCI Wellness Center shall offer incentives to students based on their participation in the app.
   2. Type: Achieve
   3. Source: 7, 9, 42
   4. Priority: Medium
10. **Access group reviews**
    1. Definition: UCI Administrators and the UCI Wellness Center shall be able to view the reviews of a group.
    2. Type: Achieve
    3. Source: 71
    4. Priority: Medium
11. **Consult group admins**
    1. Definition: UCI Administrators and the UCI Wellness Center shall be able to consult the admins of each group in order to discuss group performance and get their opinion on how performance within a group can be improved.
    2. Type: Achieve
    3. Source: 50
    4. Priority: Low
12. **Reward badges**
    1. Definition: Virtual badges (gold, silver, bronze) shall be awarded to students based on their attendance with group events. These badges will be public to everyone and displayed on a students’ profile.
    2. Type: Achieve
    3. Source: 9, 42
    4. Priority: Low

# 

Model 5: “Improve student involvement with campus resources and organizations”



Goal priorities are categorized as Low, Medium, High.

Source numbers refer to the numbered list in the Field Notes section from elicitation sessions.

1. **Improve student involvement with campus resources and organizations**
   1. Definition: Anteater Groups shall improve students’ involvement with campus resources, providing them access to resources within the app and making them aware of campus organizations.
   2. Type: Achieve
   3. Source: Case Study
   4. Priority: High
2. **Access Campus Resources**
   1. Definition: Students shall be able to access or connect to campus resources within the app.
   2. Type: Achieve
   3. Source: 26
   4. Priority: High
3. **Manage Health**
   1. Definition: Students shall be able to use campus resources to improve their mental and physical health.
   2. Type: Achieve
   3. Source: 26
   4. Priority: High
4. **Manage academics**
   1. Definition: Students shall be able to use campus resources to improve their academics.
   2. Type: Achieve
   3. Source: 26
   4. Priority: High
5. **Contact escort program**
   1. Definition: Students shall be able to call the UCI Health Safety escorts through the app when they feel uncomfortable traveling alone.
   2. Type: Achieve
   3. Source: 52
   4. Priority: Medium
6. **Contact UCI Wellness Center**
   1. Definition: Students shall be able to connect to the UCI Wellness center in order to receive mental health help.
   2. Type: Achieve
   3. Source: 26
   4. Priority: Medium
7. **Reserve Library study rooms**
   1. Definition: Students shall be able to reserve study rooms through an integrated system with UCI library.
   2. Type: Achieve
   3. Source: 43
   4. Priority: Medium
8. **Chat with academic counselor**
   1. Definition: Students shall be able to chat with an academic counselor from the app.
   2. Type: Achieve
   3. Source: [Missing Information]
   4. Priority: Medium
9. **Provide External Links**
   1. Definition: Anteater Groups shall provide embedded links to campus resources on the app.
   2. Type: Maintain
   3. Source: 52
   4. Priority: High
10. **Schedule Zoom/ Google Meet appointment**
    1. Definition: Students shall be able to schedule a Zoom or Google Meets appointment with UCI Wellness Center specialists directly through the app.
    2. Type: Achieve
    3. Source: 53
    4. Priority: Low
11. **Provide external software application**
    1. Definition: The system shall support external software applications through third party integrations so that students can connect to campus resources and organization within the application.
    2. Type: Maintain
    3. Source: 52
    4. Priority: High

## 

## **A.1.3 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 Anteater Groups. 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 Anteater Groups. 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 Anteater Groups on his phone and creates his event for the astronomy group, marking it as public and broadcasts the event to other students on Anteater Groups.

Caitlin, a 1st year UCI student and member of the Astronomy group, sees the stargazing event in her notification tab on Anteater Groups. 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 Anteater Groups 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.

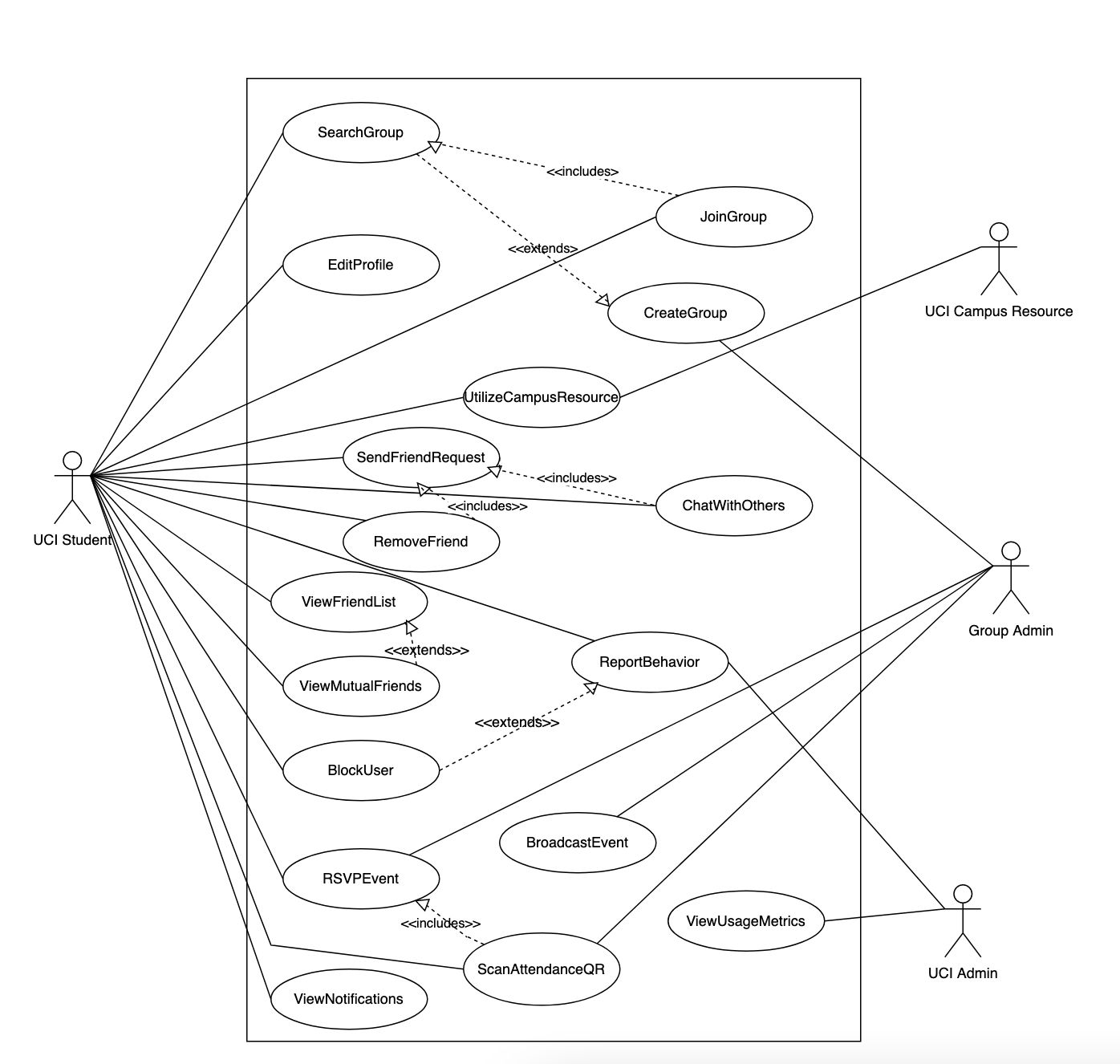
**Post-Questions:** Can students search for friends outside of their circle of mutual friends?

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 Anteater Groups app and downloaded it. She knows that her other UCI friends are also using it and already had their accounts on Anteater Groups. 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 Anteater Groups 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 A.1.3.1** | **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 A.1.3.2** | **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 A.1.3.3** | **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 A.1.3.4** | **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 A.1.3.5** | **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 A.1.3.6** | **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 Anteater Groups 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 Anteater Groups. |
| 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 A.1.3.7** | **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 Anteater Groups 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 A.1.3.8** | **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 Anteater Groups 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 A.1.3.9** | **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 A.1.3.10** | **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 A.1.3.11** | **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 Anteater Groups 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 Anteater Groups. 3. Student registers for Anteater Groups 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 Anteater Groups can RSVP for events. |
| Open Issues | N/A |

| **Section A.1.3.12** | **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 A.1.3.13** | **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 Anteater Groups such as UCI Health Safety Escorts, UCI Library study rooms, and UCI Wellness mental health specialists. They are able to access these resources within Anteater Groups, 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 Anteater Groups. |
| Failed End Condition | The student does not take advantage of an available campus resource through using Anteater Groups. |
| Trigger | The student navigates to the “Campus Resources” window in Anteater Groups. |
| Basic Flow (Main Success Scenario) | 1. The student navigates to the “Campus resources” window in Anteater Groups. 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: The 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 Anteater Groups. 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 Anteater Groups. 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 Anteater Groups 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 Anteater Groups to provide these integrated features. |

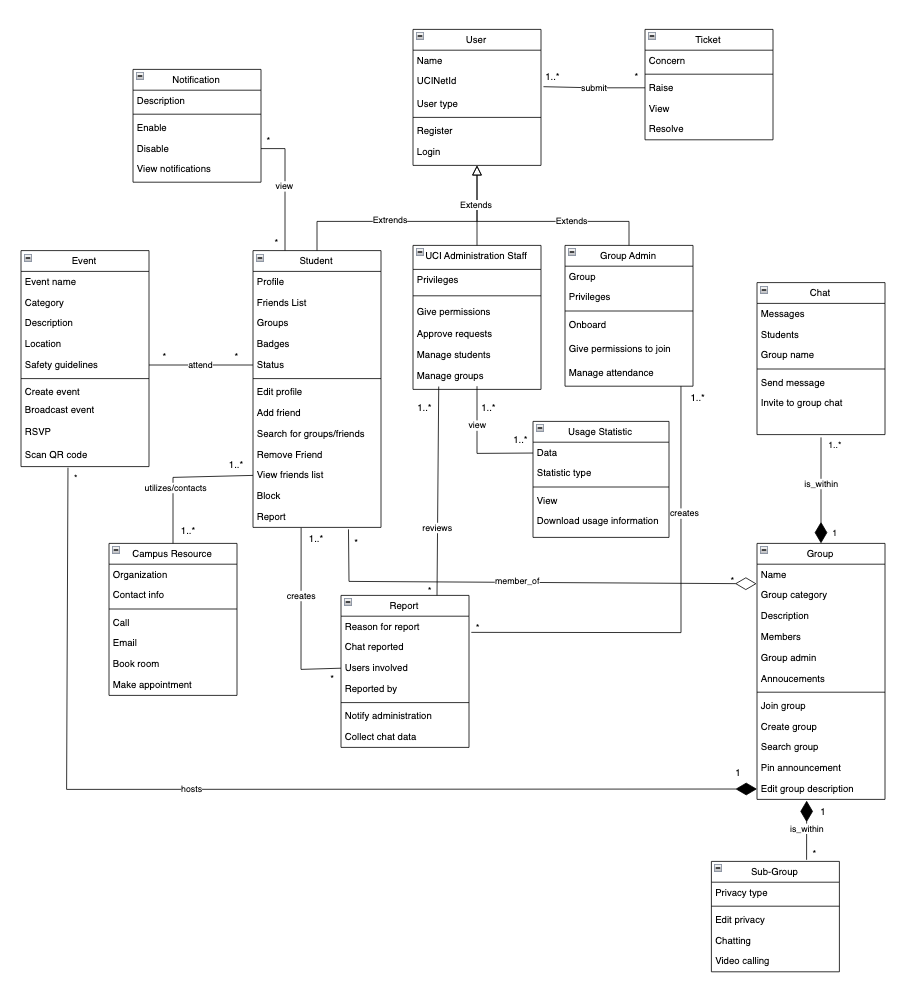
| **Section A.1.3.14** | **Content/Explanation** |
| --- | --- |
| Use Case Name | ViewMutualFriends |
| Author | Nathan |
| Priority | Medium |
| Source | Field Notes 27, 70, 79 |
| Short Description | The student are able to see mutual friends over the person’s profile/friend list  and/or list of members 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 Anteater Groups 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 Anteater Groups 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 Anteater Groups 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 A.1.3.15** | **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 Anteater Groups 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 Anteater Groups 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 Anteater Groups 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 Anteater Groups 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 A.1.3.16** | **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 Anteater Groups 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 Anteater Groups app and follow the Anteater Groups’ 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 Anteater Groups 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 Anteater Groups app and is unable to be joined by other students. The group is automatically deleted by the Anteater Groups 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 Anteater Groups app. |
| Basic Flow (Main Success Scenario) | 1. The actor(s) opens their Anteater Groups 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 Anteater Groups 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 Anteater Groups. 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 Anteater Groups 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 A.1.3.17** | **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 Anteater Groups app. A list of notifications are grouped 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 Anteater Groups 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 Anteater Groups 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 Anteater Groups 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 Anteater Groups 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. |

## **A.1.4 Domain Model: Class Diagram**

****

# 

# 

Class Descriptions

**A.1.4.1 User**

Users of the Anteater Groups include UCI students, UCI administration staff, and Anteater Groups group administrators. Operations which users can perform are register and log in. Users must register and log in to the software using their UCInetID to verify their affiliation with UCI. Upon registration, a user is categorized into one (or more) of the following categories: student, UCI administration staff, and group administrator. When a user gains another role, this shall be reflected on their profile as a badge which is visible to all other users. All users are required to use their full name as their representative/display name as registered with the UCI directory. Users are able to perform tasks such as joining and managing groups, chatting with users one-on-one, and viewing events.

Source: Field notes 16, 26, 37, 57, and 58, “Registration” goal

**A.1.4.2 Student**

A student is a person attending UCI and is enrolled in an undergraduate or graduate program. Students’ attributes include a profile, friends list, groups list, badges, and status. Operations that students can perform are: edit profile, add friend, remove friend, view friend list, search for groups/names, block, and report. Students are part of the parent *User* class, which means they inherit the attributes and operations of that found in the User class.

Source: “Improve students’ social connectedness” goal model, “Improve student involvement with campus resources and organizations” goal model

**A.1.4.3 UCI Administration**

A UCI administrator is someone who manages the students’ accounts, groups, and any activities on the Anteater Groups app. UCI admin staff has the privileges to give certain permissions to accounts on the app, approve requests from students and groups, manage students and groups and their activities, and get usage statistics data.

Source: Case study, Field Notes 38, 50, “Manage group app activity” goal

**A.1.4.4 Group Administrators**

A group administrator is a user who has unique managerial privileges for the group they are an admin of. A user becomes a group admin by being onboarded into their position. They can also give permissions to join groups and are the ones to manage group event attendance. Group administrators shall generate QR codes, promote activities, and change their group settings (public or private) at any time. Group administrators who are also graduating students shall be replaced with another student administrator.

Source: Field notes 5, 6, 8, 39, 75, 77, “Access to groups” goal

**A.1.4.5 Ticket**

A ticket is an issue or concern a User has with Anteater Groups. Tickets’ attributes include a reason for writing a ticket (the concern). Tickets can be submitted, raised, viewed, or resolved, and are accepted and handled by UCI OIT (UCI Administration).

Source: Field notes 92

**A.1.4.6 Report**

A report is performed by the group admin and students to alert misbehaving accounts on the app. Reports will be sent to UCI Administration Staff for further review and appropriate action to violated accounts; this can mean a ban on the account or a restriction for certain features on the app. More serious violations are reportable to UCI Police Department or UCI Administration for appropriate penalty for the specific student. Reports are anonymous.

Source: Field notes 12, 25, 52, 69, 72, ReportBehavior use case, “Report Inappropriate Behavior” goal

**A.1.4.7 Chat**

A chat feature will be available within groups for students to interact with one another, and group admin can participate in chat with other students as well. An algorithm will be used to scan for inappropriate messages and language in the chat for report purposes. Students can have private chat sessions with people that are on their friends list. Group admin does not have full access to chat history, and reports will be sent to group admin for further reviews and escalations.

Source: Field notes 25, 29, 33, 36, 53, 92, 93, goal model “Improve students’ social connectedness”, use case “ChatWithOthers”

**A.1.4.8 Notification**

A notification is a short description of information that the user needs to view. This could be information about friend suggestions, maintenance updates, group related events, and more. Notifications have a description and can be enabled, disabled, and viewed.

Source: Field notes 18, 41, 68, 85

**A.1.4.9 Usage Statistic**

A usage statistic provides information about the application’s use, such as headcount for a group, number of students registered, and other relevant information. Usage statistics will have data and a statistic type. UCI Administration will be able view this data and also have the ability to download this information.

Source: Field notes 38, 50, use case “ViewUsageMetrics”

**A.1.4.10 Campus Resource**

A campus resource is an organization at UCI that can be utilized by students. Students are able to call the UCI Health Safety Escorts, book a UCI Library study room, and make an appointment to meet with a UCI Wellness mental health specialist. Students can also email these organizations as their contact information is provided.

Source: UtilizeCampusResources use case, “Improve student involvement with campus resources and organizations” goal model

**A.1.4.11 Group**

Groups are made up of students who are interested in the group, and these students can chat amongst themselves within the group. Groups have a group name, a category they can be generalized under, a description that can be edited, list of members, and an admin. Groups can be joined, created, and searched for. The announcements of the group can be pinned by the group. Groups are public unless they are a sub-group, and main groups host events.

Source: Case Study, JoinGroup use case, CreateGroup use case, SearchGroup use case, Field Notes 80, “AccessToGroups” goal

**A.1.4.12 Event**

An event is a planned activity hosted by one of the groups. These events serve to encourage students to connect with each other and participate more. Events hold information about their name, category, description, location, and safety guidelines. Events can be created, broadcasted, RSVP’d, and scanned with QR codes.

Source: Field notes 5, 6, 82, 90, goal model “Enhance Students’ Interests”

**A.1.4.13 Sub-Group**

Sub-group has a composition relationship with the Group class as it is a smaller group within a group. Sub-groups can be created so that 2 or more students can chat with each other without chatting with the entire group. Students can chat and video call within sub-groups. Sub-group names are visible to all group members but accessing them requires either admin approval or a password if they are marked as private.

Source: Field notes 39, 49, 63, 93

# Team Meeting Minutes

| **Team ID: 3** | | **Date: 05.26.2022** |  | |
| --- | --- | --- | --- | --- |
| **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. Complete Front Matter and General Description areas | | We worked on these sections individually and finalized these areas at the beginning of our meeting. | |
| 1. Work on specific requirements | | We divided the work for the largest section, and also created diagrams and UI mockups. | |
| 1. Gather field notes and complete Appendix | | We completed the Appendix. | |
|  | |  | |
|  | |  | |
| **Problems encountered** | | **Resolution** | |
| 1. We were not initially accustomed to the formatting of the Functional Requirements as shown in the two examples. | We adapted to this format, as it covered everything required by the rubric. | | |
| 1. Our document is so large and with varying formatting styles that we started to have formatting issues as well as bugs. | We had to reformat everything after completion, and had to often refresh the page. | | |
|  |  | | |
|  |  | | |
|  | |  | |
|  | |  | |
| **Plans for next meeting: Activity** | | **Responsibility** | |
| 1. Continue the elicitation session on Friday and ask unanswered questions. 2. Finish Requirements section before the due date. | | All members.  All members. | |
|  | |  | |
|  | |  | |
|  | |  | |
|  | |  | |

# Missing Information

* We assumed that Anteater Groups are expected to be accessible to all UCI students and administration and accommodate any disability-related needs which impact a user’s interaction with the system using any available assistive technology.
* We assumed that Anteater Groups servers shall be able to support daily usage by the given population of UCI faculty and students.
* We assumed Anteater Groups will be using PostgreSQL and be supported on major browsers.
* What assistive technologies would you like for Anteater Groups?
* What database management system will we be using?
* Do you want the Administrator web-version of Anteater Groups to be compatible on all browsers?
* How do you want data to be encrypted?

# 

# 

# 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?
   1. Use Zoom. (forgot what else he said)
2. The reward system based on attendance
   1. We want the actual count of attendance
   2. Let’s not rely on RSVP feature (may end up not coming)
   3. Scan QR code (when they arrive) that tracks attendance, most accurate method
   4. Keep RSVP feature to know how much to expect and prepare stuff for headcount
3. How QR system works
   1. Physical: Group admin will get QR code, will be placed outside of door, attendees will scan code upon arrival
   2. 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.
   1. Student will have to request for QR code
      1. A form will populate once the QR code is scanned
6. The information for attendance
   1. The name of the student
7. Friend recommendations
   1. When registering for the first time: ‘these might be the friends you want to friend’
   2. Mail id or contact info.
   3. Compare the contact on the app with on your phone
   4. Import contact info on your phone to see friends on the app with such phone #s
   5. Mutual friends
   6. Mutual Groups (see mutual groups like on discord??)
   7. 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
   1. People’s calendar can be visible to everyone (free slot / non-free slot)
10. Event safety concerns
    1. Provide guidelines in terms of how many users can be there
    2. Detect if certain events are protests/ need UCIPD as security
    3. Check off boxes for the category of the event (entertainment, protest, etc.)
11. Have status of the user
    1. Online, offline, busy
12. Layout of groups user is in
    1. Groups they’re most active in at the top
    2. Groups in rows and columns
    3. Message bar also in top right corner
    4. About me section top left
    5. Each group has own notification bar
13. Common channel of notifications
    1. notification button top right corner
14. Can group admins give out their own personal badges or do they choose from a select few
    1. Admins can’t create their own badges
    2. 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)

## **(Week 8) - 5/20**

1. Technical tech support feature
   1. A way for ticketing or a chat support option
   2. Not able to join groups, post in groups, etc. any technical issues
   3. Raising tickets button/or chat button
   4. A separate usage statistics for OIT - tech support
   5. Rank the issues by level: P1, P2, P3 - where P1 get priority
2. Notifications
   1. Notifications for each group (within the group)
   2. General Notifications (everything else)
      1. Ads
      2. Public events
3. Event suggestions
   1. Nearby events ads on the bottom
4. Group suggestions
   1. Can be in the notifications tab/bell

## **(Week 9) - 5/27**

1. Administration see analytics on a separate web based subsystem
2. Mail/ post: can indicate a time when to be sent/posted, or if right away
   1. Ex: can post “tomorrow 9AM)
   2. ONLY feature for group admin
3. Group admins create an event, and a notification would be sent out
   1. With this they can also indicate a time for the notification to be sent
4. Send invitations for other groups/ to join events
   1. Like facebook
   2. Member of a group can “add participant” will send invitation to other user
   3. Anyone in a group can do this
   4. Private event: error if try to invite ppl who aren’t in the group
5. If a student wants to write/publish post on group, it must be approved by group admin
6. Find ways to detect bots
   1. Malicious user/hacker might be able to somehow penetrate the app/ take students’ netIDs
   2. Bots may be flooding the application with requests
   3. Implement some algorithm to ensure user is a real user/human being, not a bot
   4. We have DUO, but let’s think of more, captcha for example
7. What happens to the group when the group admin leaves?
   1. The uci admin has to look for another group admin and make the group active once again