

**CS3354 Software Engineering**  
**Final Project Deliverable 2**

utdClubs

Alex Nguyen  
Angel Villanueva  
Arhum Khan  
Azaria Rubenstein  
Cooper Kelley  
Kevin Puga  
Raul Hernandez  
Rees Blatt

## **1. Delegation of tasks**

These are the tasks that each member of our team completed from beginning to end of project.

### **Alex Nguyen**

Proposed Role: Use Case and Sequence Diagram Designer

Deliverable 1: Use Case Diagram, readme document

Deliverable 2: Test Plan and Execution

### **Angel Villanueva**

Proposed Role: Class Diagram Architect

Deliverable 1: Class Diagram

Deliverable 2: Hardware and Software Cost Estimation

### **Arhum Khan**

Proposed Role: Testing and Quality Assurance Coordinator

Deliverable 1: Sequence Diagram

Deliverable 2: Conclusion and Evaluation

### **Azaria Rubenstein**

Proposed Role: Presentation and Final Report Preparer

Deliverable 1: Compile Deliverable 1 Document

Deliverable 2: Compile Deliverable 2 Document, Presentation Slides, References

### **Cooper Kelley**

Proposed Role: Architectural Design Lead

Deliverable 1: Software Process Model, Architecture Design

Deliverable 2: Project Scheduling, Duration, and Staffing

### **Kevin Puga**

Proposed Role: GitHub Repository Setup and Documentation Manager

Deliverable 1: Created and managed GitHub

Deliverable 2: GitHub Management and Submission

### **Raul Hernandez**

Proposed Role: Software Requirements Analyst

Deliverable 1: Software Requirements, Project Manager

Deliverable 2: Cost, Effort, and Pricing Estimation, Personnel Cost Estimation

### **Rees Blatt**

Proposed Role: Project Scope and Documentation Specialist

Deliverable 1: Project Scope

Deliverable 2: Comparison with Similar Designs

## **2. Final Project Deliverable 1 Content**

Start of Final Project Deliverable 1 content.

### **1. Final Project Description**

utdClubs is an application designed to streamline the process of finding, joining, and participating in university club meetings and events. Our primary objective is to provide a platform where all university clubs can post their meeting information, which will be accessible to the student body. The application will feature a reminder system where users can choose which clubs will send notifiers of meetings and events, ensuring increased participation and engagement. Additionally, we plan to integrate the app with the UTD campus maps for easy navigation to meeting locations, enhancing the user experience by making it more convenient for students to engage with clubs they are interested in.

Our motivation for developing utdClubs stems from a recognized need within the university community for a more organized and user-friendly way to engage with clubs and organizations. Part of the hurdle of joining clubs for most students is a general lack of information, and this application will alleviate that.

utdClubs will help students jump that hurdle by offering a dynamic platform where all club-related information is consolidated and easily accessible. We hope this initiative is expected to foster a stronger sense of community, encourage greater involvement in extracurricular activities, and enhance the overall university experience for students.

In real life, we expect utdClubs being used extensively across the UTD campus, serving as the go-to source for students seeking to explore new interests, connect with like-minded individuals, and actively participate in the diverse array of clubs and organizations available.

## 2. Github Repository

### 1.1. Each team member's Github account

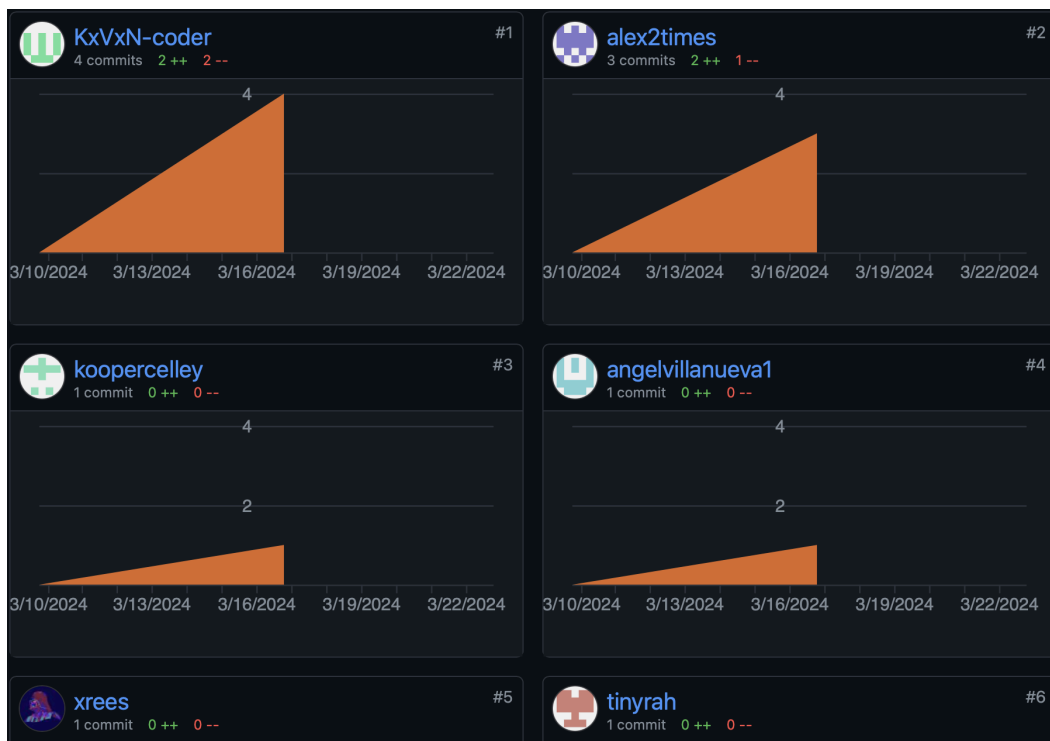
Alex Nguyen:	<a href="https://github.com/alex2times">https://github.com/alex2times</a>
Angel Villanueva:	<a href="https://github.com/angelvillanueva1">https://github.com/angelvillanueva1</a>
Arhum Khan:	<a href="https://github.com/ArhumAKhan">https://github.com/ArhumAKhan</a>
Azaria Rubenstein:	<a href="https://github.com/abr210000">https://github.com/abr210000</a>
Cooper Kelley:	<a href="https://github.com/koopercelley">https://github.com/koopercelley</a>
Kevin Puga:	<a href="https://github.com/KxVxN-coder">https://github.com/KxVxN-coder</a>
Raul Hernandez:	<a href="https://github.com/tinyrah">https://github.com/tinyrah</a>
Rees Blatt:	<a href="https://github.com/xrees">https://github.com/xrees</a>

### 1.2. Create team Github repository

<https://github.com/KxVxN-coder/CS3354-utdClubs>

### 1.3. Add all team members to Github Repository

Proof of team members' activity:



### 1.4. Make the first commit to the repository

README file:

<https://github.com/KxVxN-coder/CS3354-utdClubs/blob/main/README.md>

## 1.5. Project Scope

[https://github.com/KxVxN-coder/CS3354-utdClubs/blob/main/project\\_scope.pdf](https://github.com/KxVxN-coder/CS3354-utdClubs/blob/main/project_scope.pdf)

### Objective

- 1.1 To enhance social integration among UTD students by connecting them with clubs and events via a dedicated app.
- 1.2 To transform UTD's "commuter school" reputation by fostering a vibrant campus community.

### Features

- 2.1 User Registration
  - 2.1.1 Open for individual students and club representatives
  - 2.1.2 Options to join as a club member or as an individual without affiliations
- 2.2 Interest Profiling
  - 2.2.1 Users provide their interests to tailor event recommendations
- 2.3 Event Discovery
  - 2.3.1 A personalized "For You" landing page displaying current and popular events
  - 2.3.2 Club-specific pages showcasing respective events to enhance club visibility
- 2.4 Account Customization
  - 2.4.1 Customizable profiles including major, year, interests, and social media links
- 2.5 Communication
  - 2.5.1 Direct messaging between students and club organizers for event inquiries
- 2.6 Event Tagging
  - 2.6.1 Use of tags for easy event discovery based on time, location, and other preferences
- 2.7 Campus Map Integration
  - 2.7.1 Interactive UTD map featuring event locations, club meeting spots, and key campus facilities
  - 2.7.2 Real-time updates and navigational assistance to event venues within the campus

### Benefits

- 3.1 A centralized platform for all student organization activities, casual meet-ups, and large social events
- 3.2 Facilitates student engagement, especially during free time, by providing an accessible guide to campus events
- 3.3 Promotes new campus facilities, like the esports arena, by highlighting associated events
- 3.4 Enhances navigational ease and spatial awareness on campus with an integrated map feature

### Target Audience

- 4.1 Primarily UTD students and faculty for enhancing campus life and community engagement

## 1.6. Team Github repository

<https://github.com/KxVxN-coder/CS3354-utdClubs>

### 3. Delegation of tasks

#### For deliverable 1:

This is what each member did for the assignment due 03/22/2024.

**Alex Nguyen**

Did: Use Case Diagram, readme document

**Angel Villanueva**

Did: Class Diagram

**Arhum Khan**

Did: Sequence Diagram

**Azaria Rubenstein**

Did: Compiled Final Project Deliverable 1 Document

**Cooper Kelley**

Did: Software Process Model, Architecture Design

**Kevin Puga**

Did: Created and managed GitHub

**Raul Hernandez**

Did: Software Requirements, Project Manager

**Rees Blatt**

Did: Project Scope

We tweaked our original task delegation slightly, however, it is still accurate to everyone's roles:

1. GitHub Repository Setup and Documentation Manager - **Kevin Puga**
2. Project Scope and Documentation Specialist - **Rees Blatt**
3. Software Requirements Analyst - **Raul Hernandez**
4. Use Case and Sequence Diagram Designer - **Alex Nguyen**
5. Class Diagram Architect - **Angel Villanueva**
6. Architectural Design Lead - **Cooper Kelley**
7. Testing and Quality Assurance Coordinator - **Arhum Khan**
8. Presentation and Final Report Preparer - **Azaria Rubenstein**

#### **4. Which software process model is employed in the project and why?**

The software process model that utdClubs will use is the incremental method. We plan to use this model because it allows for the most flexible way in which we can complete the development of utdClubs. We believe this because it allows us to more easily adapt to changes in requirements, especially because we are not limited to any developmental barriers, as we are a small team of developers. It allows us to create software versions that align with our goals at a specific stage of development, while still allowing the ability to reiterate previous ideas as specifications and requirements change. Additionally, it allows the utdClubs team to work closely together in the development process, which will then create a better final product.

## **5. Software Requirements**

### **5.a. Functional Requirements**

#### **1. User Account Management:**

System will allow users to create and manage their accounts. This includes:

- Registering with an email and password

- Editing their profile information (major, year, interests, social media links)

- Deleting their account

#### **2. Event Creation and Management:**

System will allow CLUB accounts to create and manage events. This includes:

- Setting event details (title, description, date, time, location, tags, dietary restrictions)

- Editing existing events

- Deleting events

#### **3. Event Search and Filtering:**

System will allow users to search for events based on keywords, tags, date, time, location, and dietary restrictions

#### **4. Event Recommendation:**

System will recommend events to users based on their interests and past event attendance

#### **5. Messaging:**

System will allow users to send and receive messages to and from users and organizers

### **5.b. Non-functional Requirements**

#### **1. Product Requirements:**

Performance: App shall load the "for you" landing page with new and popular events within an average of 2 seconds

Reliability: App shall have an uptime of 99.5% during peak hours which would be weekdays between 10am and 4pm

Availability: App shall be available on all platforms (iOS, Android, Web)

Security: App shall use secure user authentication and data encryption to protect user info

Maintainability: App shall be designed in an iterative and modular way to allow for easy maintenance and updates

Portability: App shall be designed to be easily ported to other university campuses, without needing a major change in the framework

#### **2. Organizational Requirements:**

Standard Compliance: App shall comply with UTD's security policies and accessibility standards

Development Process: App shall be developed using an iterative process to systematically add features. This will allow us to quickly put up our MVP.

#### **3. External Requirements:**

Interoperability: App shall be able to integrate with existing UTD systems, like campus map and event calendar

Privacy: App shall comply with all applicable privacy laws and regulations, (i.e. FERPA)

Ethical Considerations: App shall be designed to promote inclusivity and diversity, and to avoid bias in event recommendations

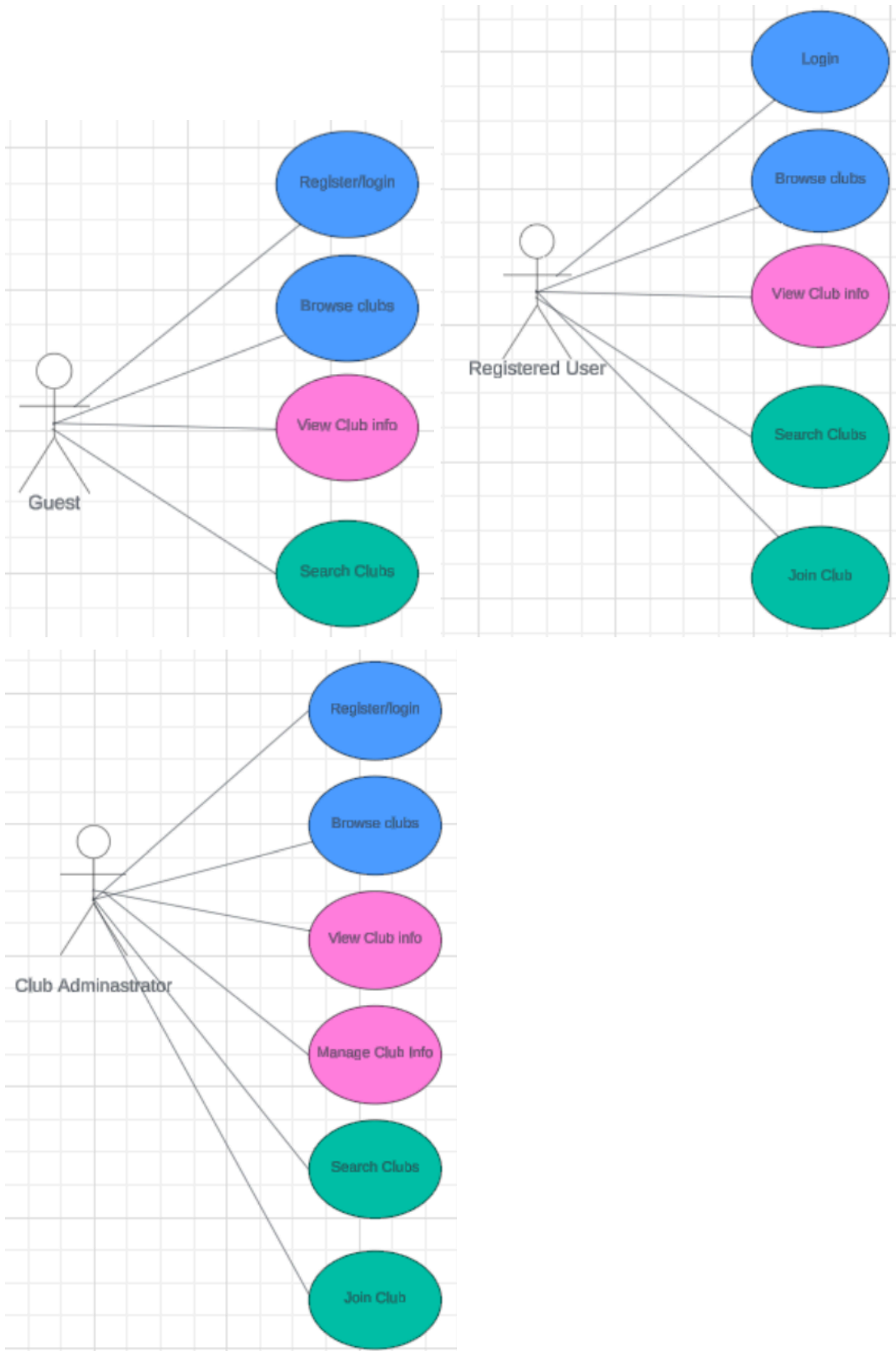
#### **Assumptions:**

When developing the app, we might need to make assumptions about some external factors, like the availability of APIs for existing UTD systems.

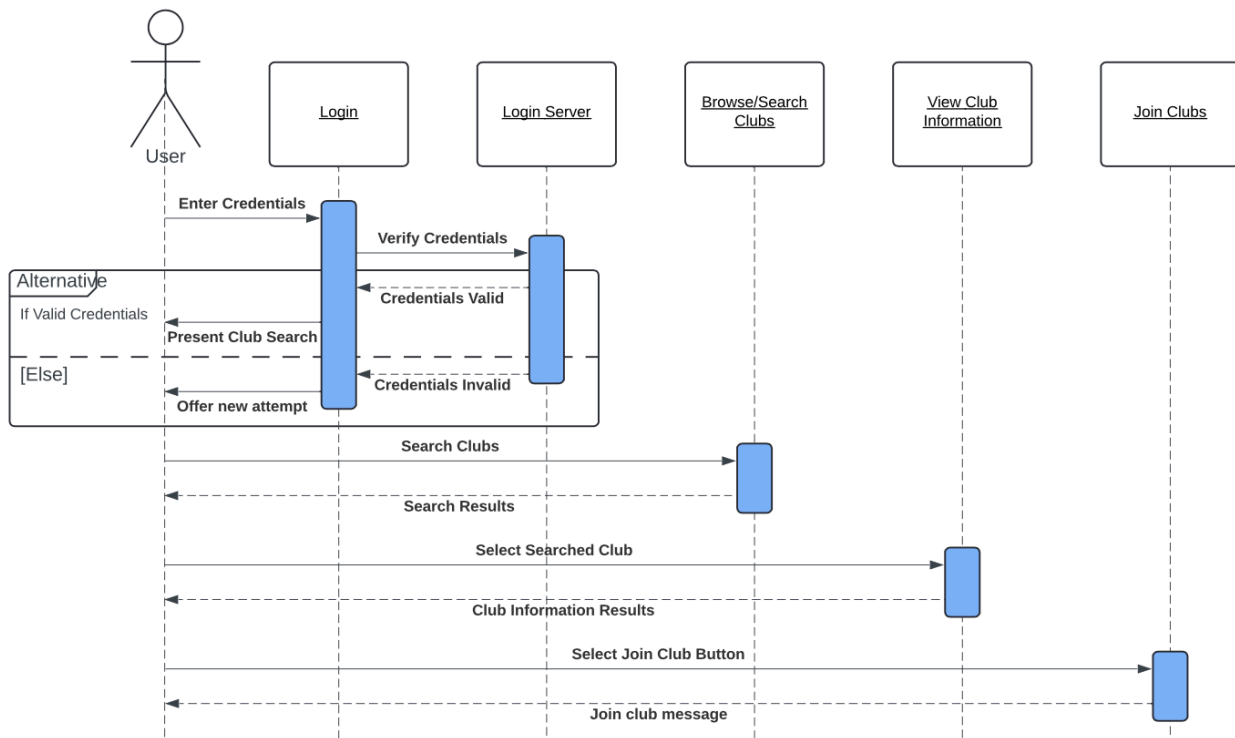
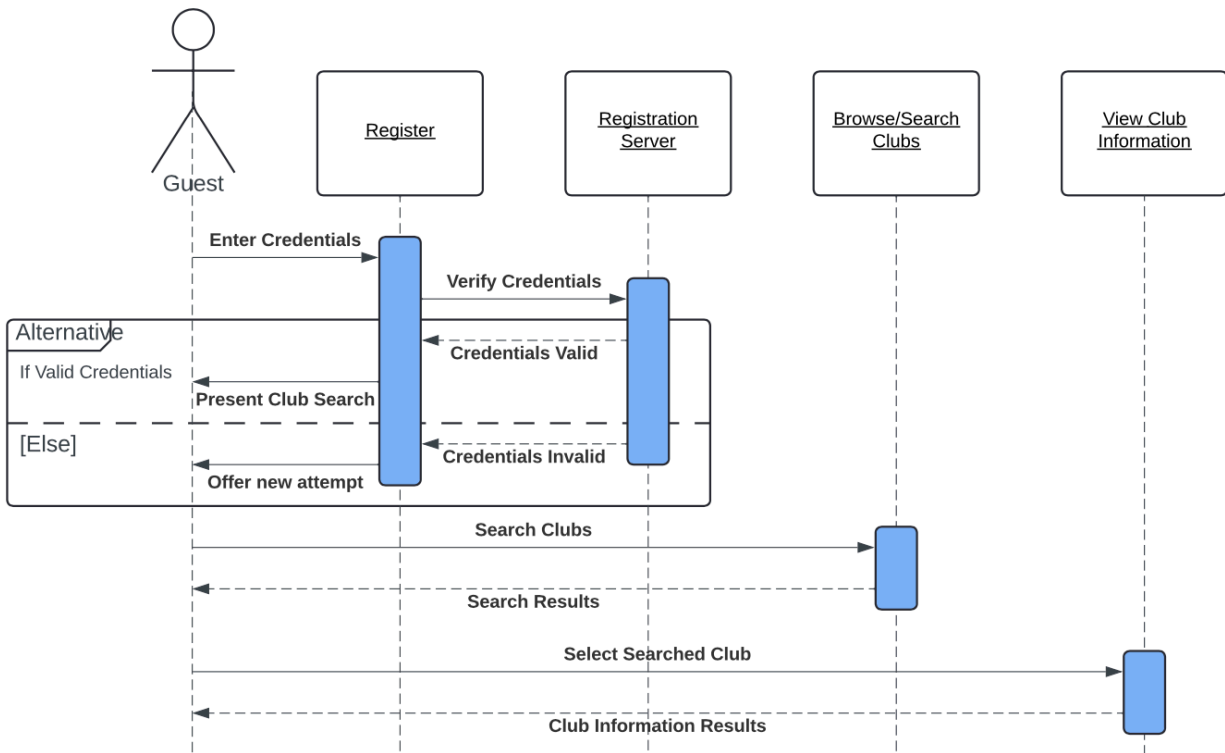


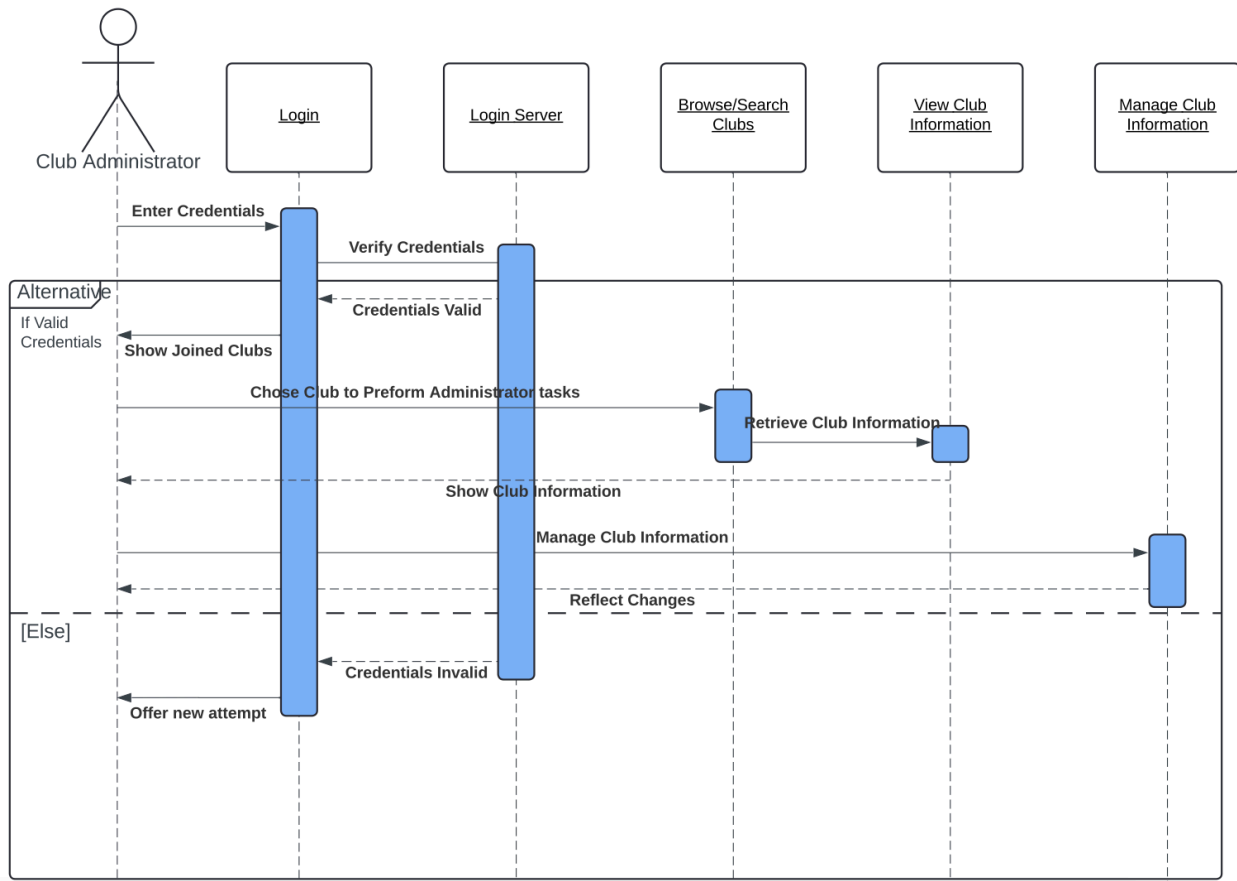
## 6. Use Case Diagram

REVISED:

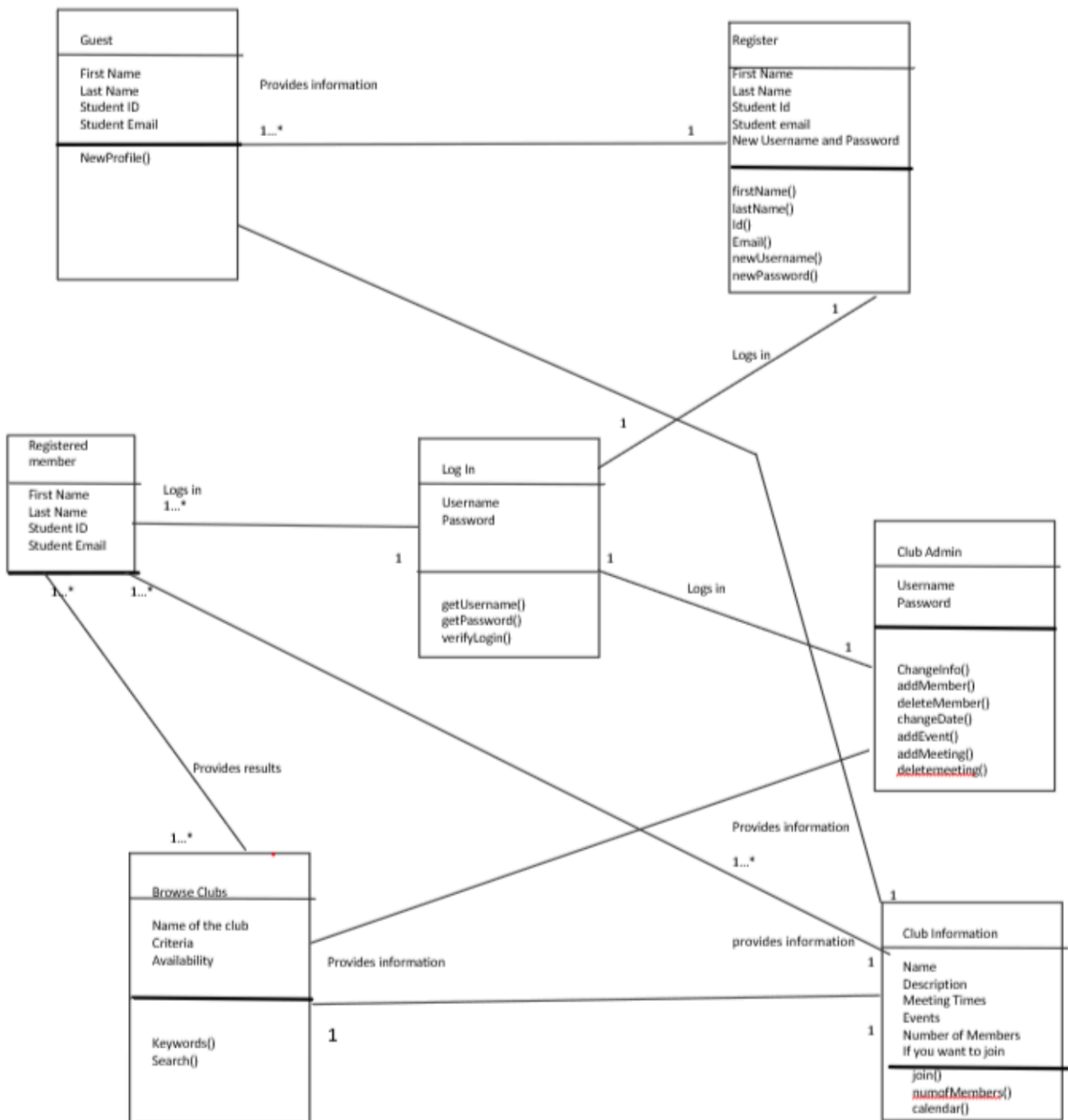


## 7. Sequence Diagram

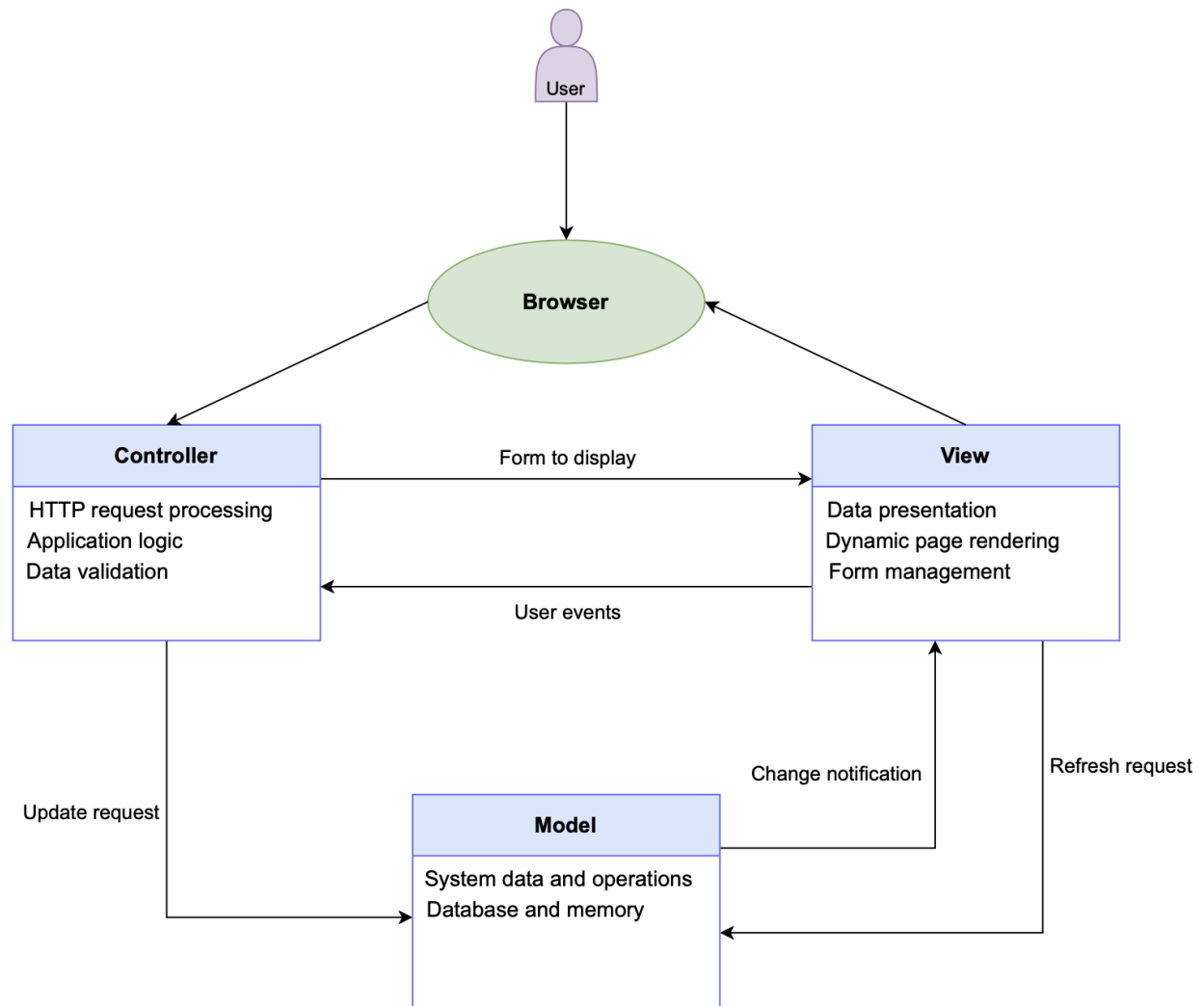




## 8. Class Diagram



## 9. Architectural Design



End of Final Project Deliverable 1 content.

### **3. Project Scheduling, Cost, Effort, Pricing Estimation, Duration, and Staffing**

#### **3.1 Project Scheduling**

Estimated Project Duration: 10.5 months

Justification: Based on Function Point (FP) estimation, we assume the project will take about 8.5 months. However, we propose an additional 2 months to account for the following:

- Customer and internal meetings regarding the project
- Intensive internal testing
- Customer demonstrations
- Live Implementation and testing
- Other possible unforeseen issues that may arise
- Time off (both employees and customers)

Note that this duration is tentative and could change based on the points above.

Proposed Project Start Date: June 1, 2024

Justification: After proposing this project to an organization, upper-level management will likely want to talk it over, allocate resources such as who will be working on the project, and let engineers finish any other projects that they are currently working on.

Note that this start date is tentative and could change to earlier or later depending on how long the above issues take to be sorted out.

Proposed Project End Date: April 14, 2025

Justification: Based on the previously stated estimated project duration and proposed project start date, we propose that the project will be finished and live for customers on April 14, 2025. Besides the estimated project duration, this proposed project end date also considers the following:

- Weekends
- Mandatory holidays provided by the company and the client
- PTO and/or other reasons that engineers and/or management may be out of the office or unable to work on the project.

Weekends and Hours Per Day:

We do not include weekends in our estimate, as our company uses a 40-hour work week estimate. However, depending on individual performance, each engineer and engineering management is subject to abiding by working a minimum of 40 hours per week, regardless of the days per week, or hours per day that they may work.

Note that this policy is tentative and subject to change based on the following:

- Work Performance o Required meetings
- Other issues that may arise

### 3.2 Cost, Effort and Pricing Estimation

#### Assumed User Inputs:

1. Registration (Email, password, first name, last name) [Simple, 4]
2. Profile editing (major, year, interests, social media, profile picture) [Simple, 5]
3. Event Creation (title, description, date, start time, end time, location, tags, dietary restrictions) [Simple, 9]
4. Event editing (same as above but update) [Simple, 9]
5. Event search (keyword search, tag selection, date selection, time selection, location selection, dietary restriction selection) [Average, 6]
6. Messaging (compose message, select recipient) [Average, 2]

Simple = 27

Average = 8

$$\text{GFP\_inputs} = (27 * 3) + (8 * 4) = 81 + 32 = 113$$

#### Assumed User Outputs:

1. User profile display [Simple, 1]
2. Event details display [Simple, 1]
3. Event search results [Simple, 1]
4. Messaging inbox [Simple, 1]
5. Messaging conversation view [Simple, 1]

Simple = 5

$$\text{GFP\_outputs} = (5 * 4) = 20$$

#### Assumed User Queries:

1. Event search by keyword [Simple, 1]
2. Event search by tag [Simple, 1]
3. Event search by date [Simple, 1]
4. Event search by time [Simple, 1]
5. Event search by location [Simple, 1]
6. Event search by dietary restriction [Simple, 1]
7. User search [Simple, 1]
8. Club search [Simple, 1]

Simple = 8

$$\text{GFP\_queries} = (8 * 3) = 24$$

Assumed number of data files/relational tables:

1. User profiles [Simple, 1]
2. Events [Simple, 1]
3. Clubs [Simple, 1]
4. Messages [Simple, 1]

Simple = 4

$$\text{GFP}_{\text{data}} = (4 * 7) = 28$$

Assumed number of external interfaces:

1. Integration with UTD campus map [Average, 1]
2. Integration with UTD event calendar [Average, 1]

Average = 2

$$\text{GFP}_{\text{external}} = (2 * 7) = 14$$

$$\text{GFP} = 97 + 20 + 24 + 28 + 14 = 199$$

Processing Complexity:

1. (4)
2. (4) App requires integration with UTD campus map and event calendar
3. (0)
4. (3) Not expected to have much load, so performance is of average importance
5. (0)
6. (4) Most of the app's functionality revolves around online data entry like registration
7. (3) average transaction rate
8. (5) Need real-time updates for events, profiles, messaging
9. (1) no significant complexity
10. (2) no significant complexity, only thing would be event recommendations
11. (3) some components can be reused, like user auth and event management
12. (0)
13. (2) designed specifically for UTD but can be ported to other universities
14. (3) should be able to accommodate future change

$$\text{PC} = 4 + 4 + 0 + 3 + 0 + 4 + 3 + 5 + 1 + 2 + 3 + 0 + 2 + 3 = 34$$

PCA:

$$\text{PCA} = 0.65 + (0.01 * 34) = 0.99$$

FP:

$$\text{FP} = \text{GFP} * \text{PCA}$$

$$\text{FP} = 199 * 0.99 = 197.01$$

Effort:

$$\text{Effort} = \text{FP} / \text{productivity} = 197.01 / 8 = 24.626 \text{ person-months}$$

Our team size is 8 so project duration is

$$\text{D} = 24.626 / 3 = 8.208 \text{ months}$$



### **3.3 Hardware cost estimation**

Our group will be using VERCEL to host our server. They have a free plan that will suffice all our needs if we are not using it for commercial use. Since we won't be making any money from this as it is a simple platform for UTD clubs, it will be very useful to us. Essentially, our hardware costs will be \$0.

### **3.4 Software cost estimation**

For licensing, we would have to pay \$99 to Apple to be able to distribute it on the Apple App Store, and would be free for Google Play.

For API,

- Google Calendar API for scheduling clubs will cost \$0
- Google Maps to locate clubs will cost \$0

### **3.5 Personnel cost estimation**

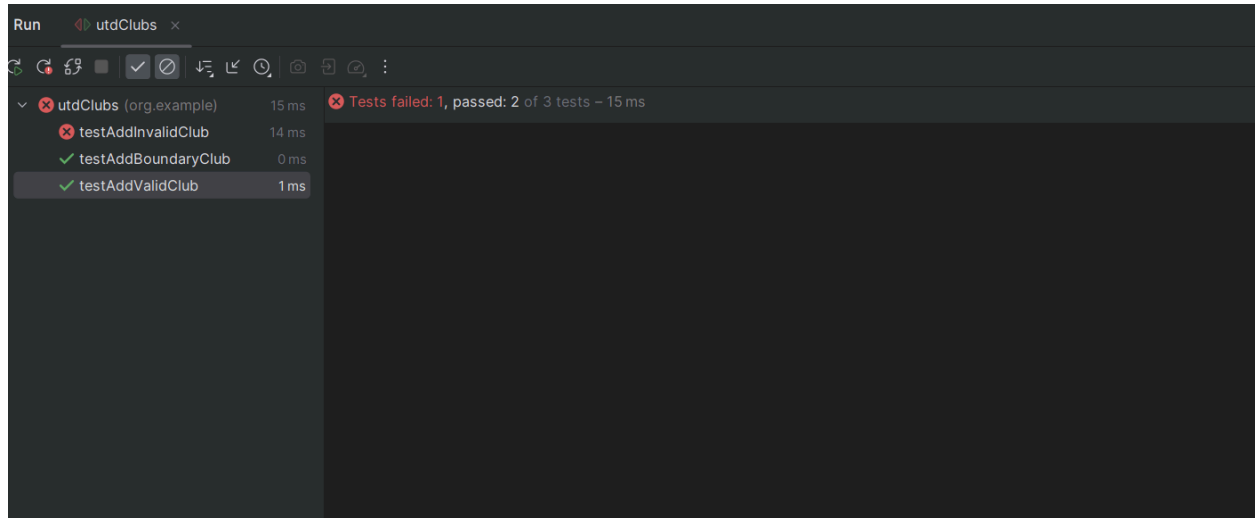
Three developers could implement this. Each would have a salary of 5,000 a month and since we estimate the project would take 8.208 months for 3 developers which comes out to \$123,000. Training cost would be only \$100 as there would be a maximum of one hour of training for this software.

## 4. Software Test Plan

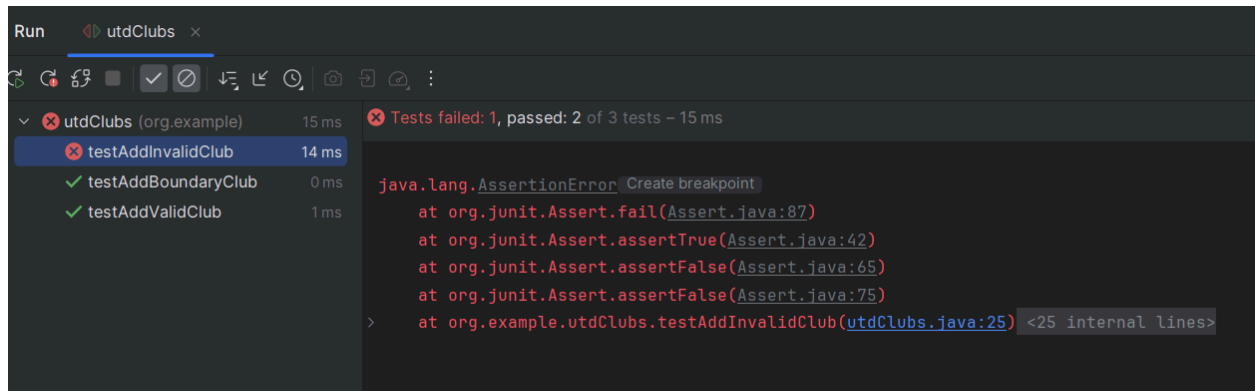
This software test plan tests to see if the user has a valid input for the club name. If they do not input a name, it will fail the case. We used JUnit testing.

Java code attached to Final Project Deliverable 2 submission as a .java file.

Screenshot of JUnit valid test:



Screenshot of JUnit invalid test:



## **5. Comparison with Similar Designs**

The most similar piece of software to our current project would be the app Cove. It is an app created with the goal of cataloging events as they take place on different college campuses. Essentially, a digitized flier system where students can sort by events taking place on different days across the campus they attend. The app also has a live chat feature where students can comment under different digital fliers advertised on the app. This differs from our app which focuses more on select campuses such as UTD. We have a deeper searching criteria that allows students to sort events not only by date and time but also by location, club, and dietary restriction(s). This allows for a deeper functionality where students are more certain that events they are looking at are ones they would be interested in attending. Our biggest paradigm shift would be in that our app is catered more specifically to the UTD population, taking into account all buildings in campus and schools such that events are more precisely sorted. The sorted algorithms used in Cove are more arbitrary, really only taking into account the date and time of the event taking place, and not much more. Also, we found the live chat functionality that Cove employs under digital fliers to be somewhat superfluous. Since we really did not see a reason for that integration, we focused on other aspects of our application instead. As we feel that fundamentally the most important part of any app seeking to connect students, is to be as accessible as possible with extremely specific searching criteria such that students can discover events they would be interested in attending as painlessly as possible.

## **6. Conclusion**

utdClubs is an application designed to enhance the university club experience by simplifying the process of finding, joining, and participating in club meetings and events. By acting as a centralized platform where university clubs can post their schedules and updates, utdClubs makes an active university life accessible to students. The app also integrates with campus maps to help users easily locate meetings, aiming to foster a stronger community and enrich the overall university experience at UTD.

With our UTDClubs application, we aim to develop event registration, event creation, and messaging methods while following all organizational and external requirements and considerations. To achieve this, we are still using the incremental method as we do not have developmental barriers and would allow us to adapt to changes during the developmental process. While revising our project, we decided to fix our previously incorrect use case diagram. Other than changing the diagram, the team decided that no further changes were necessary for the development and potential implementation of our project.

## 7. References

[1] J. Dinh, "Cove - College Events," *Apple App Store*, Mar. 13, 2024. Available: <https://apps.apple.com/us/app/cove-college-events/id1632046016>. [Accessed: Apr. 19, 2024]

[2] J. Dinh, "Cove College Events," *www.covecollegeevents.com*, 2022. Available: <https://www.covecollegeevents.com/>. [Accessed: Apr. 19, 2024]

## **8. Presentation Slides**

Presentation slides attached to Final Project Deliverable 2 submission as a .pptx file.