



chatmaps

Nick Pease, Stephen Goodridge, Clark LaChance, Joseph Gallant, Aidan Bradley

25 April 2024



Overview



- Web-based social networking service
 - www.chatma.ps
- Connect with others in your local geographical area
- Features an interactive map showing general location relative to users + rooms
 - Open chat rooms to start public conversations based on specific topics
- Registered users can create their own chat room
 - Create topics of interest for others to view or join
- Features direct messages between users

Our aim is community engagement, meaningful interactions, and a new way to meet others



User Stories



- As a registered user, I want to be able to view chat rooms near me so that I can talk to people in my community.
- As a registered user, I want to be able to create my own chat room so that I can talk with friends in private.
- As a registered user, I want to be able to include my personal interests in my profile so that I can find like minded people.
- As a registered user, I want to be able to appear offline so that I won't be disturbed.
- As a registered user, I want to be able to filter my search for chat rooms by using keywords so that I can find where I belong.



User Stories



- As a registered user, I want to be able to add other users so that I can reach out to them in the future.
- As a registered user, I want the ability to delete previous messages that I sent in the case that I mistakenly wrote to the wrong chat.
- As a registered user, I want to receive chat notifications so that I can respond back promptly to friends.
- As a registered user, I want to be able to send direct messages to other users so that I can chat with them in private.
- As a registered user, I want to be able to view who's online so that I know who I can chat with.



Kanban Board

ChatMaps ☆ Workspace visible Board

Power-Ups Automation Filters

- User Stories To Do
- User Stories - Doing
- User Stories - Done
 - As a registered user, I want to be able to filter my search for chat rooms by using keywords so that I can find where I belong.
 - As a registered user, I want to be able to paste GIFs in chat so that I can reference funny memes.
 - As a registered user, I want the ability to delete previous messages that I sent in the case that I mistakenly write to the wrong chat.
 - As a registered user, I want to be able to set language filters for my chat rooms so that I can limit explicit language.
 - As a registered user, I want to be able to view who's online so that I know who I can chat with.
- To Do - Deliverable 5
- Done - Deliverable 5
 - Updated Github
 - Software
 - Updated Kanban Board
 - Updated Product Backlog
 - Use Case Models
 - Architectural Design
 - Domain Model
 - Design Pattern/DCD
 - Configuration
 - Updated Sprint Backlog 5
 - Plan for Future Document
 - SRS (Update Pics)

Clark

Sprint Backlog



ID	User Story	Story tasks	Sprint number	Sprint goal	Story type	Story Points	Story Points Completed	Names and Contribution Percentage
7	As a registered user, I want to be able to appear offline so that I won't be disturbed.	Implement offline functionality to chatrooms/friends list	5	User will be offline in the chat when they leave the chatroom	F, T	2	2	Nick, 33%, Clark, 33%, Aidan, 33%
		Appear offline if app is closed	5	When user closes app, appear offline to friends	NF, T, KA	1	1	Nick, 100%.
		Code profile page for user	5	All user profiles, Show online/offline statuses on profile page	F, T, KA	2	2	Nick, 25%, Stephen 25%, Clark, 25%. Aidan, 25%
12	As a registered user, I want to be able to view who's online so that I know who I can chat with.	Design UI for friends list	5	Mockup UI for friends tab	F, KA	1	1	Aidan, 20%. Clark, 20%. Stephen, 20%. Joe, 20%. Nick, 20%
		Implement adding friends	5	Allow user to add friends and show their online status	NF, T	1	1	Nick, 33%. Clark, 33%. Aidan, 33%.
		Implement online status of friends	5	Show user who's online on their friends list	F, T	1	1	Nick, 100%.
9	As a registered user, I want to receive chat notifications so that I can respond back promptly to friends.	Design UI for notifications	5	Show user a notification when a chats been updated	F, KA	1	1	Aidan, 20%. Clark, 20%. Stephen, 20%. Joe, 20%. Nick, 20%
		Implement notification UI	5	Add UI to notify user of chat updates	F, T	1	1	Nick, 50%. Clark, 50%.
		Implement in app notifications	5	Show user notifications for incoming chats or chatroom updates	NF, T	2	2	Nick, 50%. Clark, 50%.



Product Backlog

Completed User Stories:

1. **As an unregistered user, I want to be able to view chat rooms near me so that I can talk to people in my community.**
2. As a registered user, I want to be able to include my interests in my profile so that like minded people can find me.
3. **As a registered user, I want to be able to create my own chat room so that I can talk with friends in private.**
4. **As a registered user, I want to be able to add other users so that I can reach out to them in the future.**
5. As a registered user, I want the ability to delete previous messages that I sent in the case that I mistakenly write to the wrong chat.
6. As a registered user, I want to be able to filter my search for chat rooms by using keywords so that I can find where I belong.
7. As a registered user, I want to be able to appear offline so that I won't be disturbed.
8. As a registered user, I want to be able to see a short list of users when I hover over a chat so that I can see if I know anyone.
9. As a registered user, I want to receive chat notifications so that I can respond back promptly to friends.
10. As a registered user, I want to be able to paste GIFs in chat so that I can reference funny memes.
11. **As a registered user, I want to be able to send direct messages to other users so that I can chat with them in private.**
12. As a registered user, I want to be able to view who's online so that I know who I can chat with.
13. As a registered user, I want to be able to set language filters for my chat rooms so that I can limit explicit language.
14. As a registered user, I want to be notified when a new user joins the chat so that I can greet them.
15. As a registered user, I want to be able to logout of my account at any time so that I can keep my messages secure.
16. As a registered user, I want to be able to visually see how big a chat room is so that I can avoid busy chat rooms.

(Bolded stories indicate stories with particular importance to the usability of the application)



Functional Requirements



- The system shall let users create chat rooms with a specific topic, for other users in the area to join.
- The system shall show users a local map of their area.
- The system shall have the option to create private chat rooms which can only be accessed by invitation.
- The system shall keep users personal information on their account private.
- The system shall allow users to customize their profile.



Non-Functional Requirements



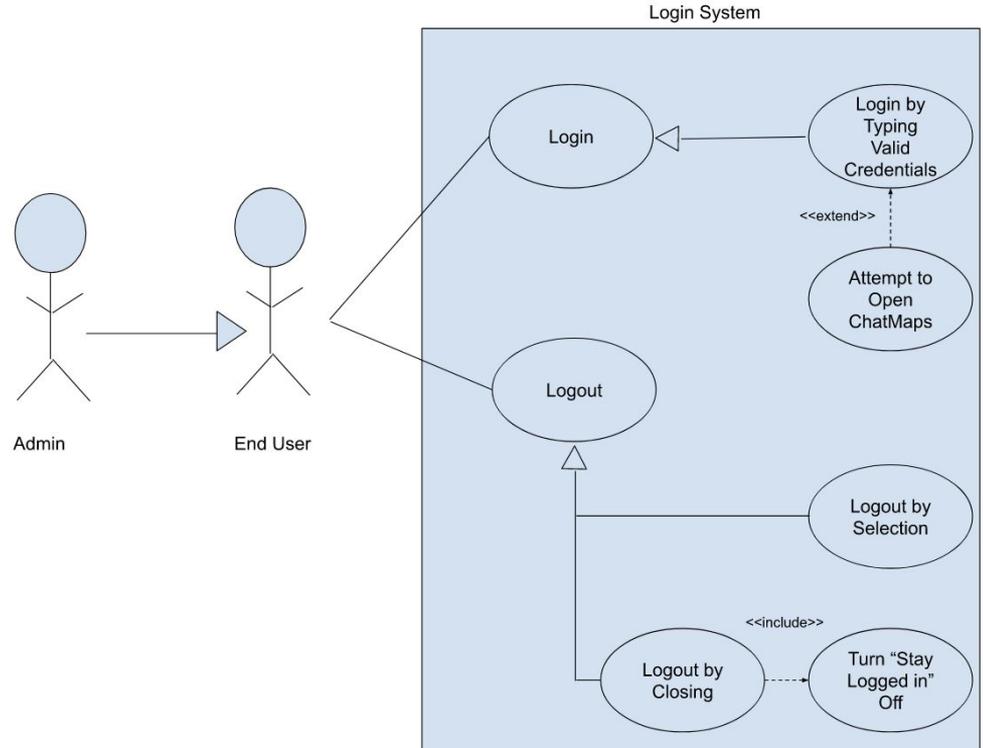
- The system shall be available at all hours of the day 99% of the time.
- The system shall provide users real-time location updates every 5 minutes 95% of the time.
- The system shall be capable of keeping users' information private 99% of the time.
- The system shall comply with relevant laws and regulations, such as data protection regulations 99% of the time.
- The system shall handle sending messages within 5 seconds 95% of the time.



Use Case 1

Login System

- Logging in is a requirement to access main features of ChatMaps System.
- It's important to handle logins and logouts appropriately without security risks.
- When logging out, terminating the current session is ideal to prohibit others from accessing your account on the same computer.
- Logging in can be done by providing the correct credentials that matches the information on our server.
- Logging out can be done by clicking the logout button or simply by closing the browser.



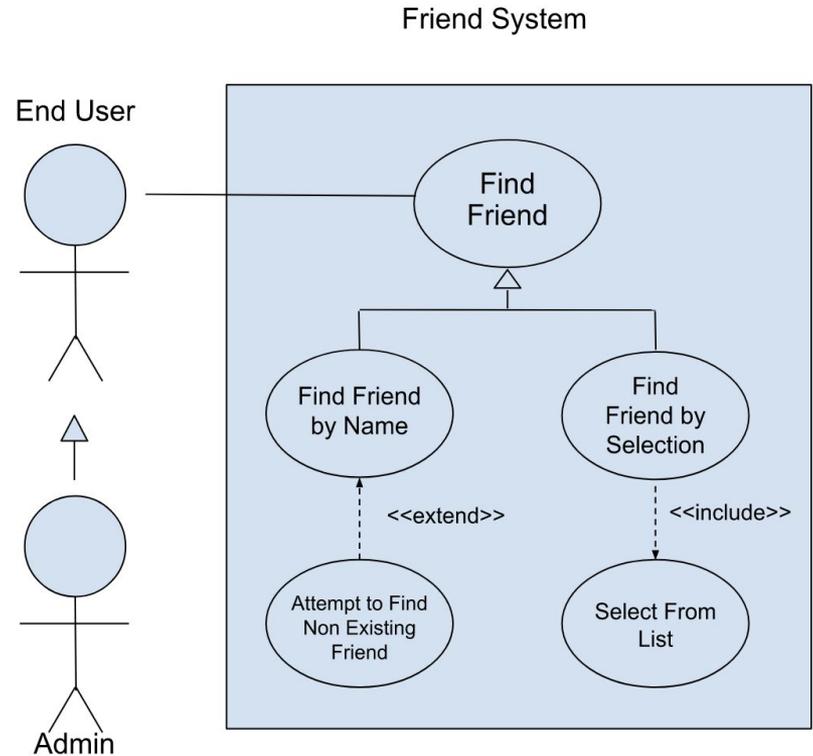


Use Case 2



Friend System

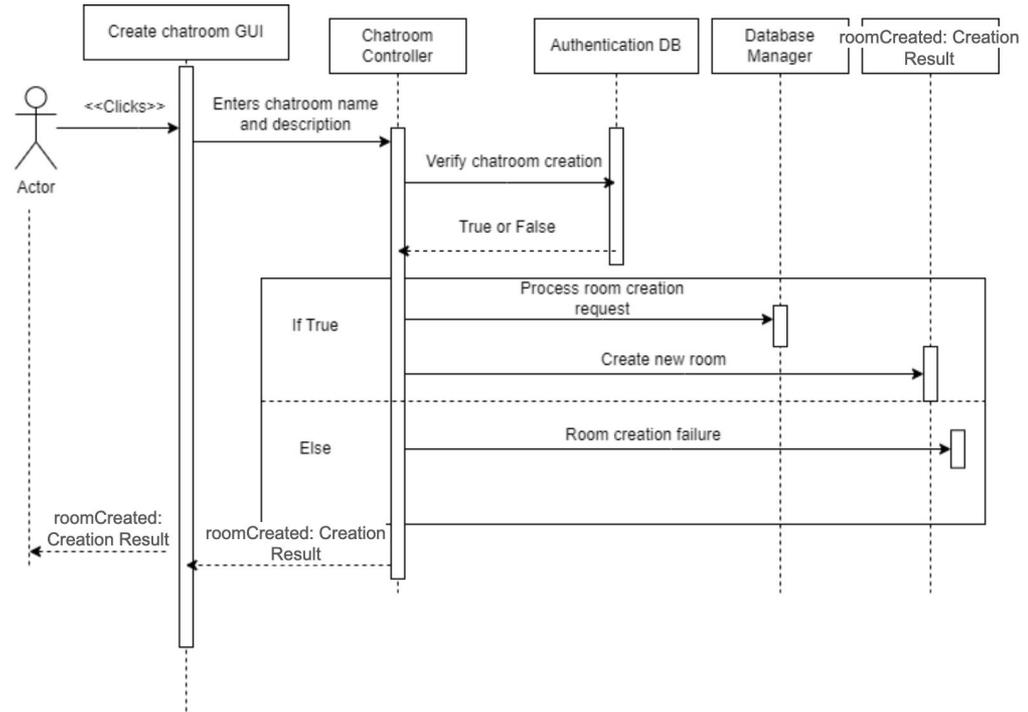
- Friends are an important part of any social network and can increase user engagement and retention.
- This exact friend system was not implemented completely as of 25 APR 2024. Friend lookup by username is a planned feature for forward releases.
- Adding a friend can be done by accessing the users profile and pressing the Add Friend button.
- In this use case, you may lookup a username to see if it exists, or select one from a list of all users



Sequence Diagram 1



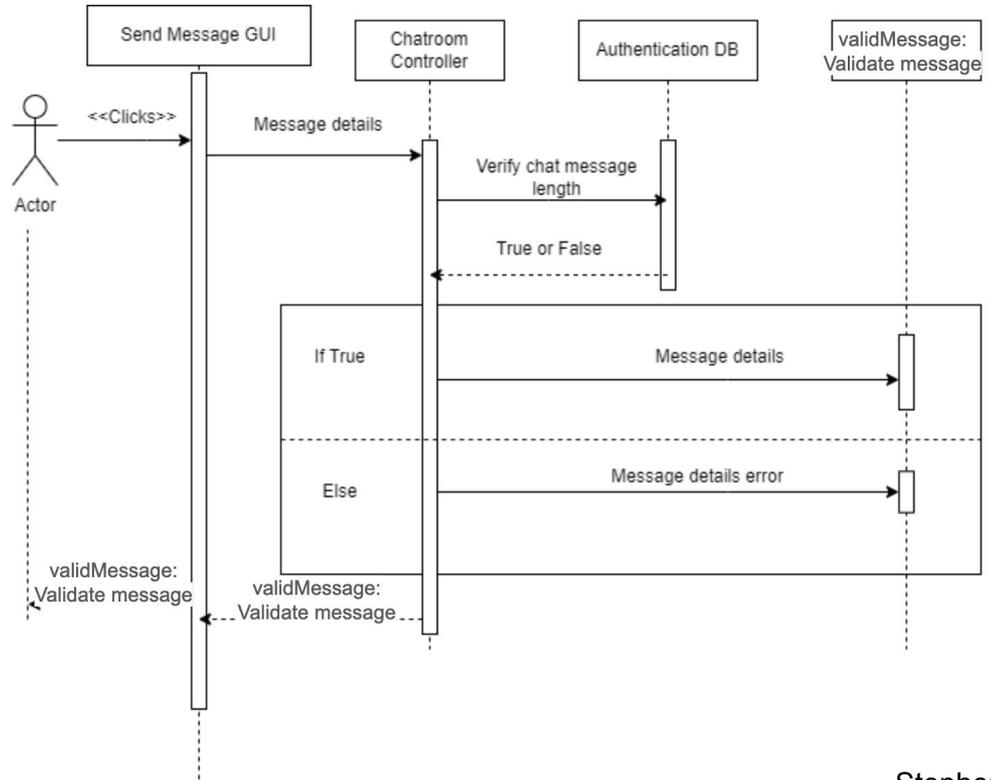
- Creating a chat room, while a benign feature, requires a fair amount of computation and verification as to prevent unauthorized tampering with the application, spam room generation, or overload of DB or Authentication services.
- Every time a chat room is created, the user is authenticated to ensure currency on its current authorization
- If authorization succeeds, the room is created and stored in the proper place in the database
- If authentication fails, the user is redirected to the authentication flow to process a new authentication.





Sequence Diagram 2

- Sending a message is crucial to the fundamental operation of this software.
- Sending a message primarily involves verifying that the length of the message is in accordance with the max storage size of the database
- Once that verification succeeds, the message is uploaded to the database and distributed to all clients currently connected to its corresponding chatroom
- The user can confirm this by ensuring that the message is evident on their screen.

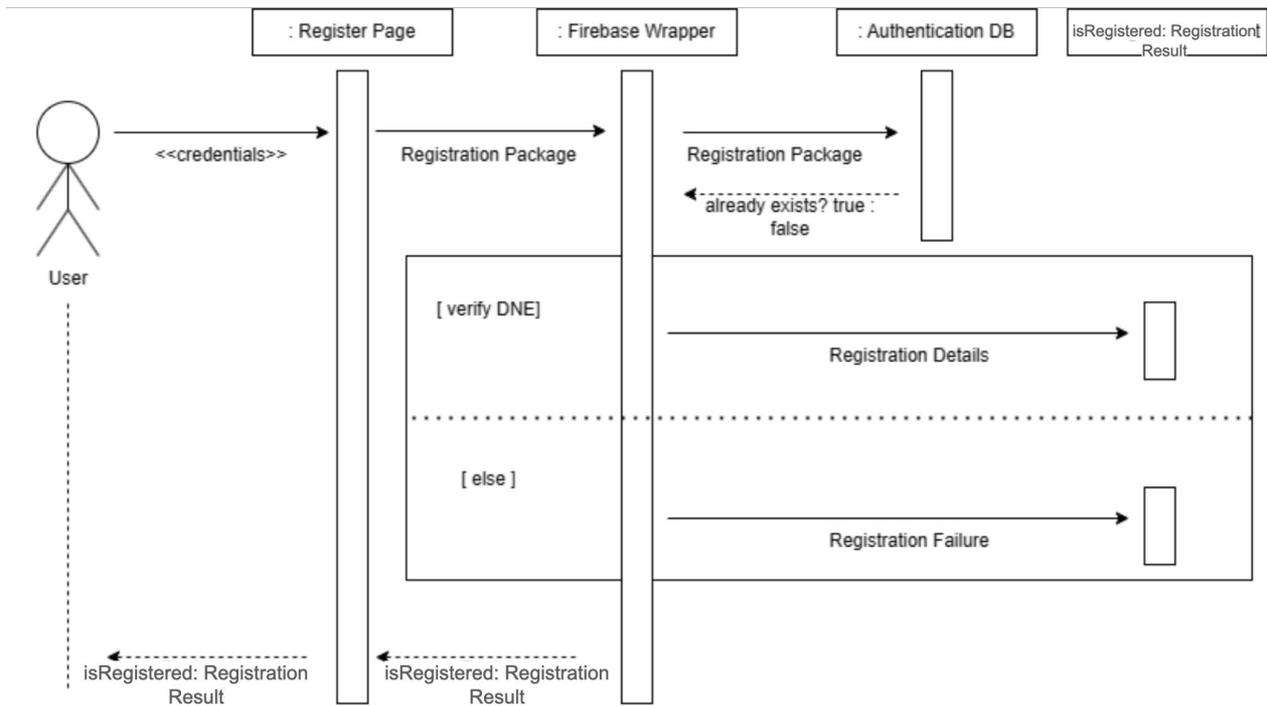




Sequence Diagram 3

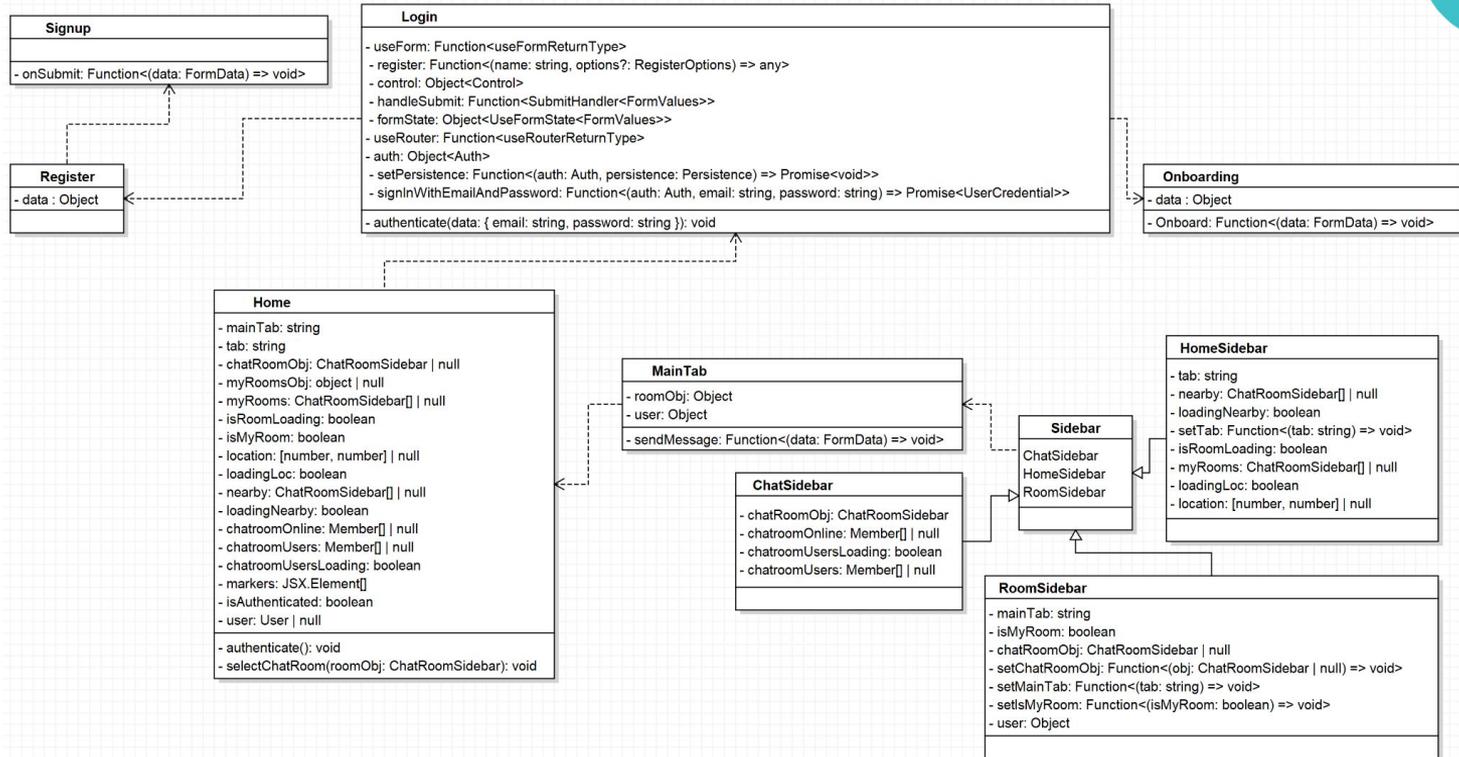


- Registration





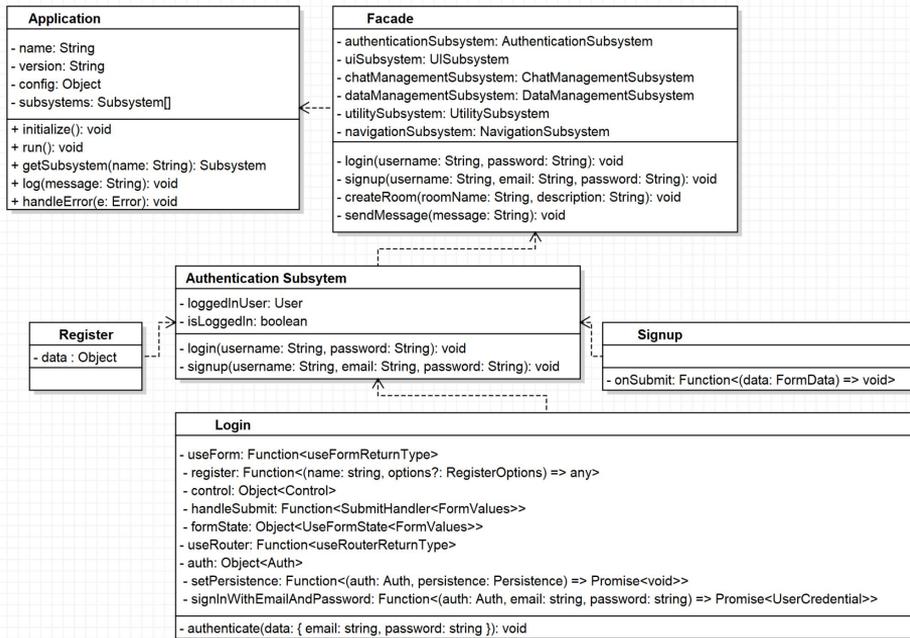
Domain Model



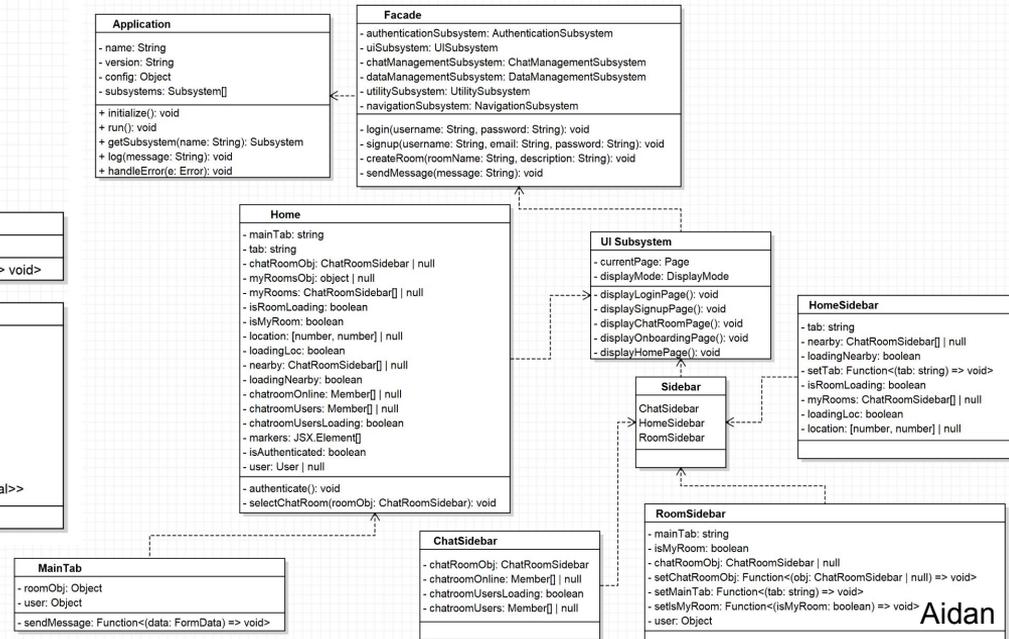


Design Class Diagrams

Authentication



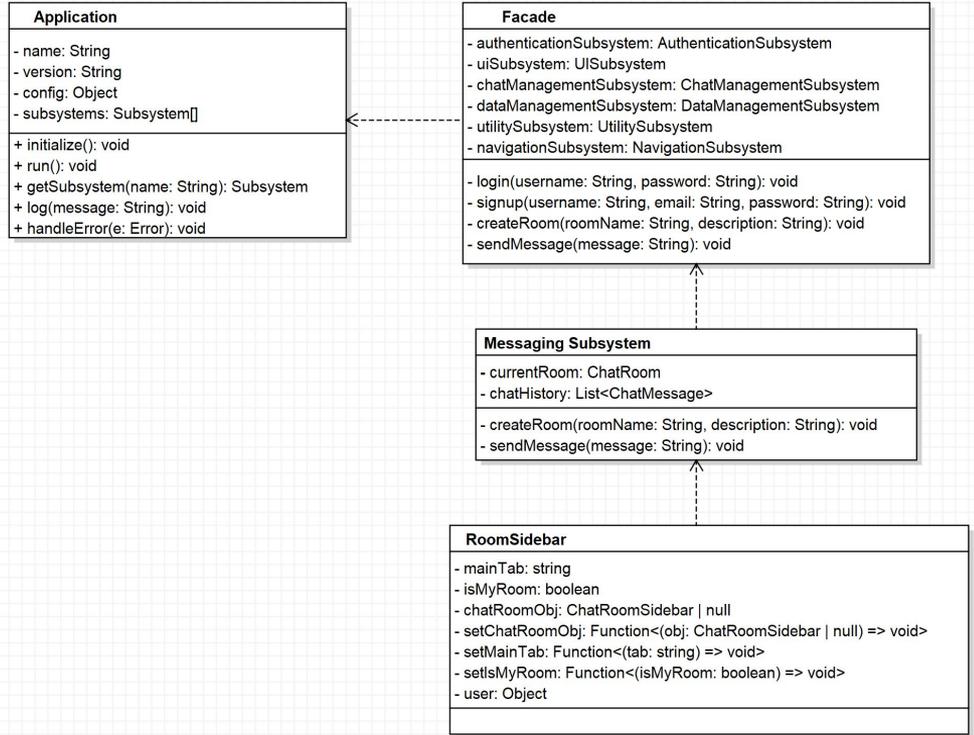
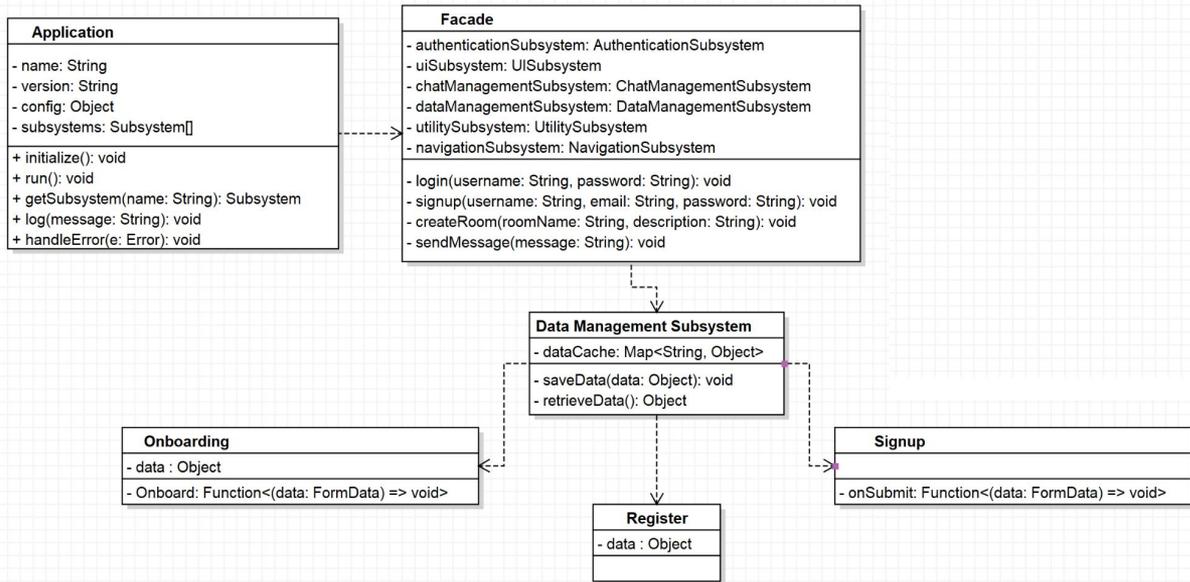
User Interface



Design Class Diagrams

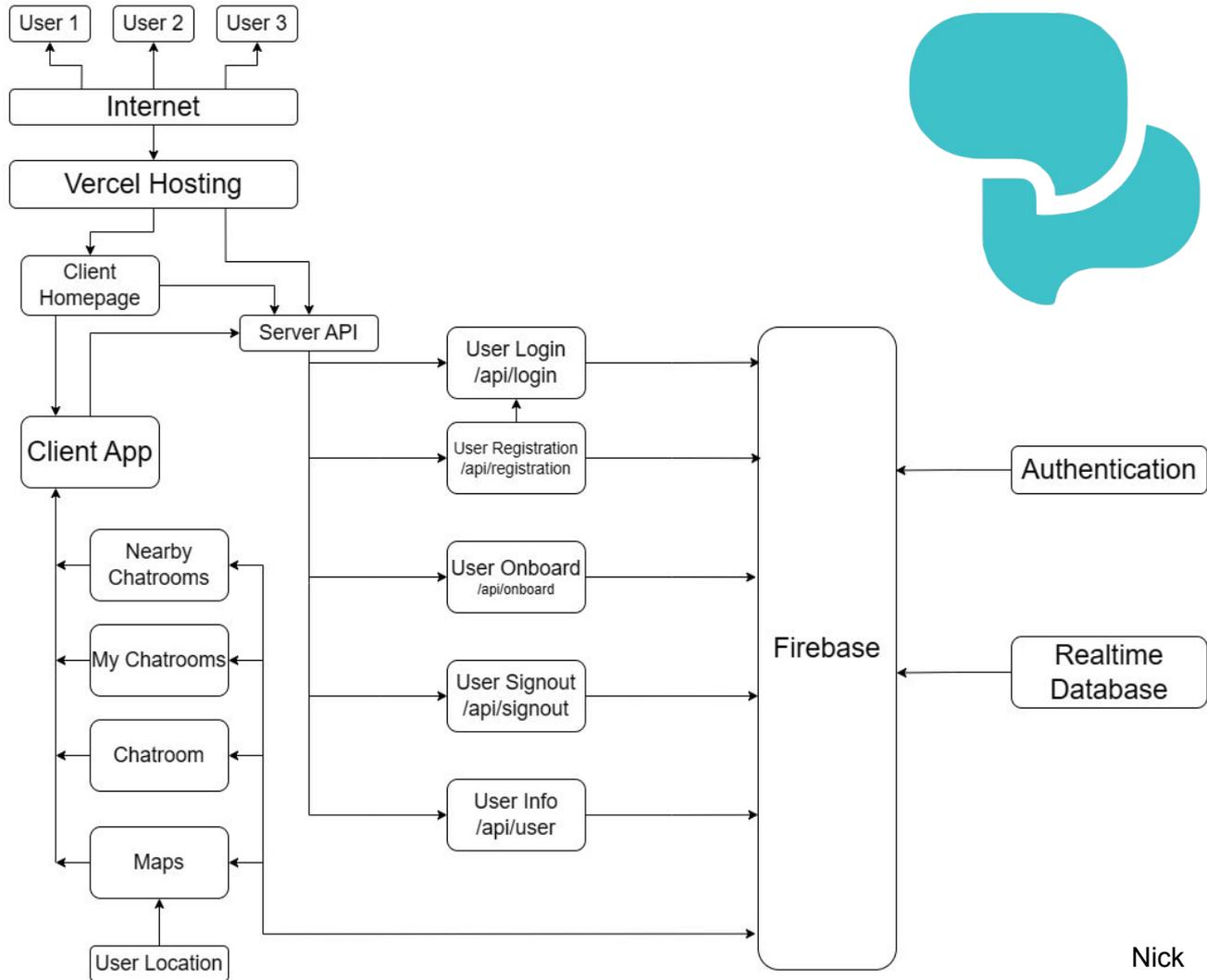


Data Management



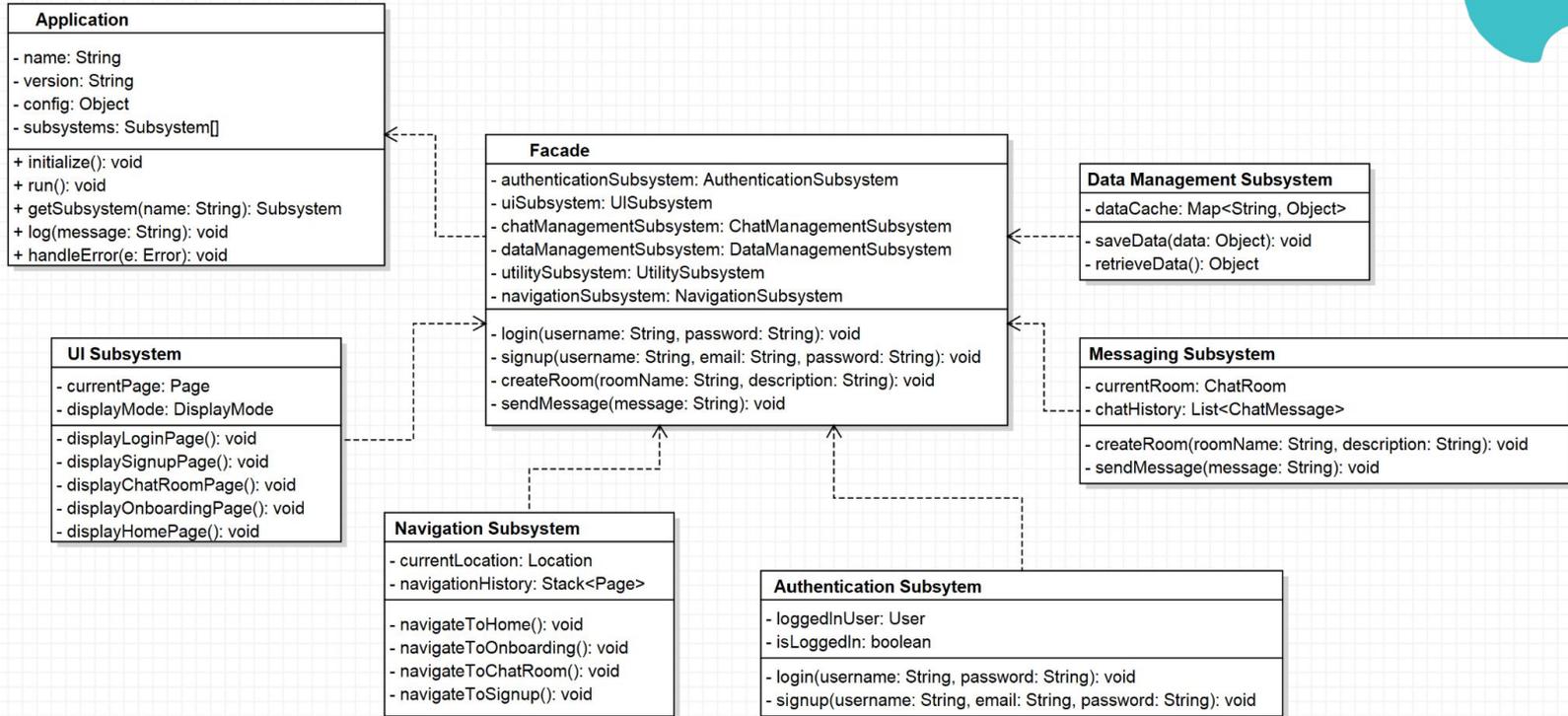
Messaging

Architecture





Architecture (Design Pattern)





Configuration Management

Github Repository: <https://github.com/ChatMaps/ChatMaps>

Current Version: *V1.0.1 (Usability Patch 1)*

Release Scheme (Incremental Model)

[major release] . [minor release] . [bug/security patch]

Key Tools Used

- Github (Distributed Control Management System)
 - Vercel (Preview, CI/CD, Web Application Distribution)
- Firebase (Database, Authentication, File Storage)
 - React (Frontend, Backend)



Test Plan

Use Case Tests

1. Login by Typing Valid Credentials
2. Add Friend by Selection
3. Create a Room
4. Accept a Friend Request
5. Join a Room

Test Plan

Unit Tests

Test Case 1 (Code) Object: Chat	Test Case 2 (Code) Object: ChatRoomSidebar
Test Case 3 (Code) Object: DM	Test Case 4 (Code) Object: Member
Test Case 6 Object: Friend	Test Case 7 Object: FriendRequest
Test Case 8 Object: Geo	Test Case 9 Object: Notification
Test Case 10 Object: HomePage	

Test Case 5 (Code)

Object: RMF (Rich Message Formatting)

Input:

Message {String} - String containing text, urls, and images

Output:

**[] React Components and Strings*

Example Implemented Unit Test Case

```
import { RMF } from "../src/components/app/datatypes" ;
import renderer from 'react-test-renderer' ;

var message = "Hello, World! This is a test message.
https://www.google.com"

it('RMF Renders Correctly' , () => {
  const tree = renderer
    .create ((RMF (message)))
    .toJSON () ;
  expect (tree).toMatchSnapshot () ;
});
```

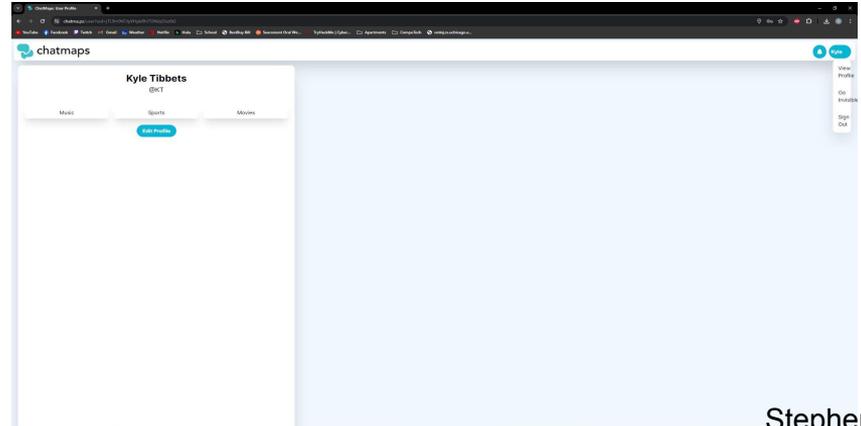
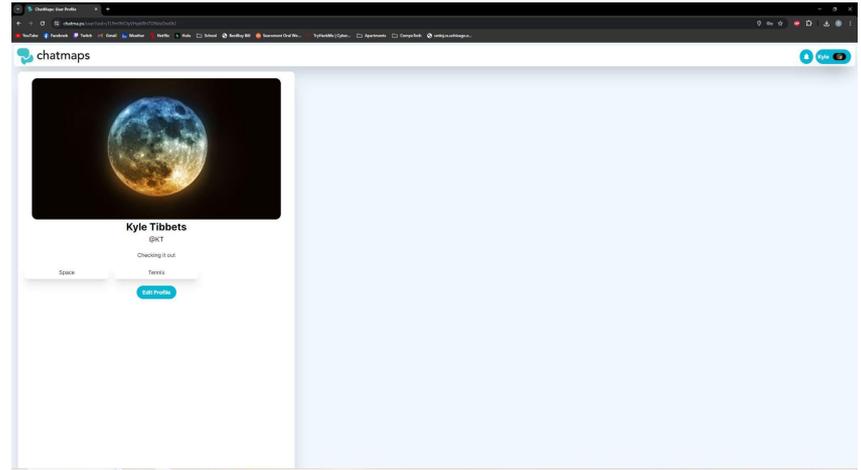
Test Plan

— Acceptance Testing

User: Kyle Tibbets

Feedback:

- Easy to navigate UI
- Very customizable profile page
- Liked the GIF feature in chat rooms
- Small bugs around dropdown menu size when no profile picture is added.



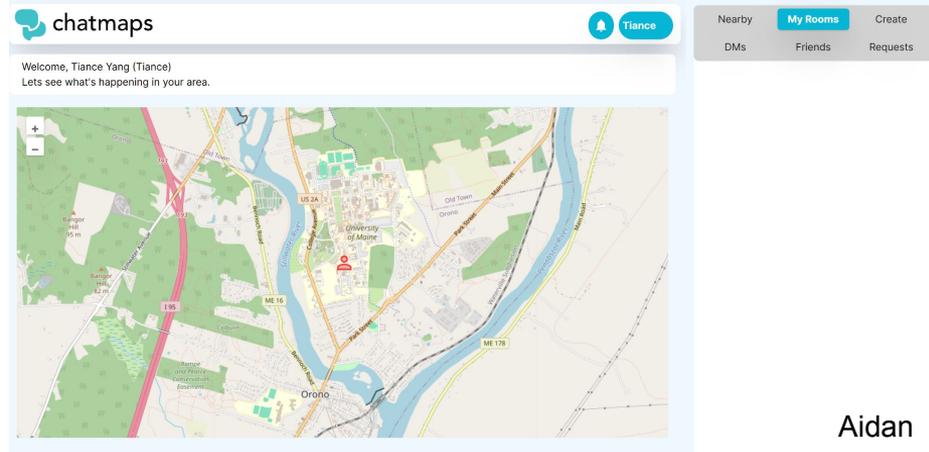
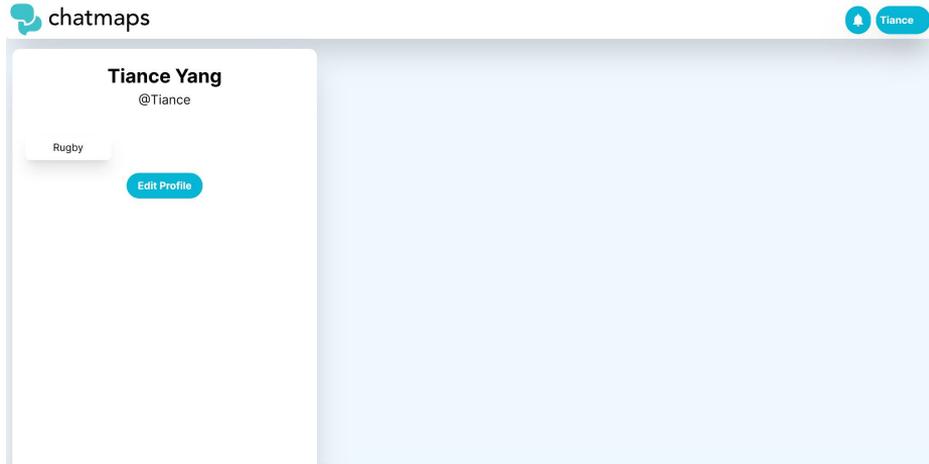
Test Plan

Acceptance Testing

User: Tiance Yang

Feedback:

- Easy to create an account
- Nice features, intuitive interface
- Map feels empty. Could do more on it
- The chats are really smooth and work well





Security & Privacy

- Security and privacy are major concerns while implementing a social platform such as ChatMaps
- As such, the entire application (save for the first page) utilizes Firebase authentication services to handle keeping track of the users authentication across tabs and the site.
- Each page individually verifies authentication status is current before completing connection
- On very important pages such as profile editing and DM's, additional checks are implemented to verify the correct users are accessing those pages / elements.
- Database access is governed at a server level by access restrictions, restricting users from editing information they are not authorized to, in addition to preventing unauthorized deletions.
- Profanity filters are implemented on the chat side, preventing most lewd or profane messages from being shown to the user.

```
service firebase.storage {
  match /b/{bucket}/o {
    match /{document=**} {
      allow read: if request.auth.uid != null
      allow write: if false
    }
    match /users/{userID}/pfp {
      allow read: if request.auth.uid != null
      allow write: if request.auth.uid == userID
    }
  }
}
```

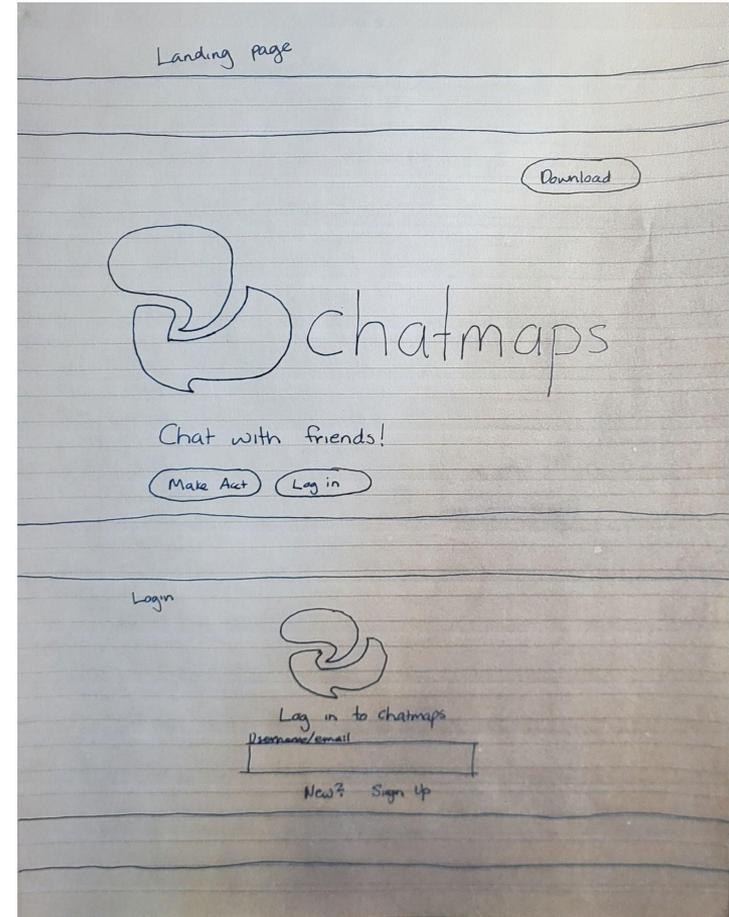
Example of Firebase access restriction rules for the profile picture storage backend

Mockups (1/3)



Chat with friends!

Login Sign Up



Mockups (2/3)



chatmaps

Chat with friends!

Register

Enter Email Address

Enter Password

Re-enter Password

Register

Have an account? [Log In](#)

Sign - Up

Create your chatmaps account

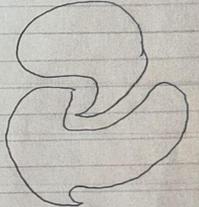
Email address

Username

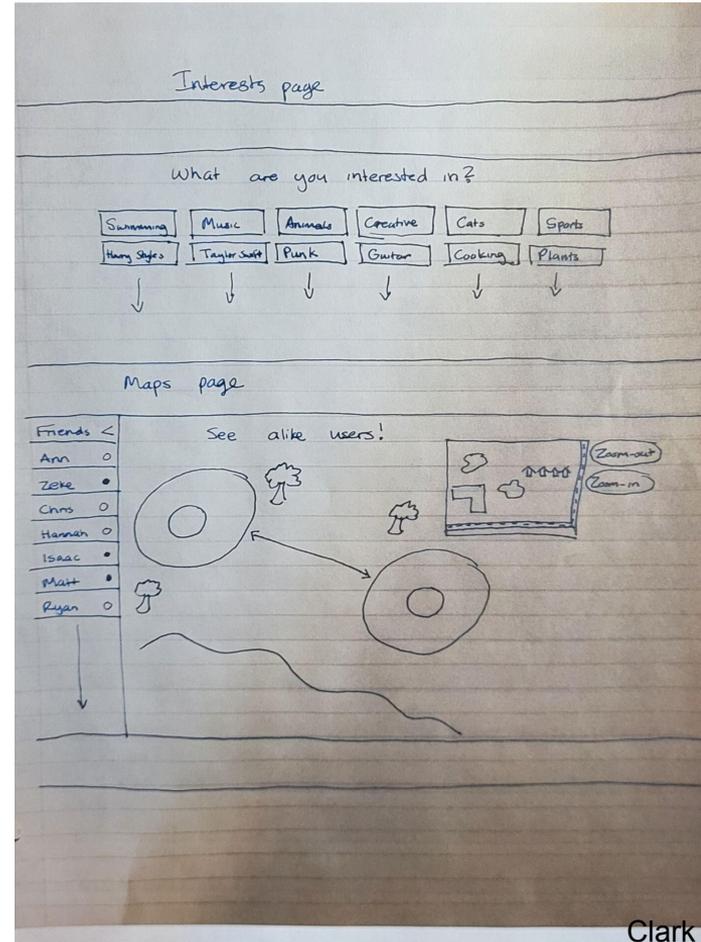
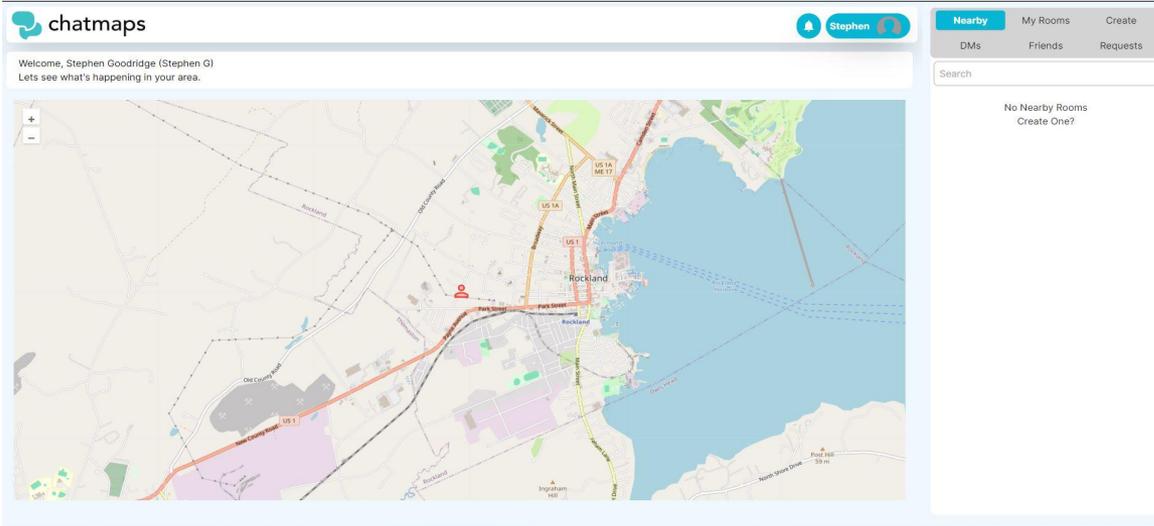
Create a password

Keep me signed in

[Terms of Use / Privacy policy
acknowledgement?](#)



Mockups (3/3)





Workload Distribution



Nick - 30%

Clark - 20%

Stephen - 20%

Aidan - 15%

Joe - 15%



Future Plan



- Chatmaps team finished all the user stories we set out to do
- While we've reached our goals, there's still some room for improvement and refinement.
- The team has plans to enhance the user experience and focus on quality of life improvements that won't disrupt the functionality of our system.

Examples:

- Ability to lookup users by username for easy friend requesting
- Add features to the map to help it not feel so 'empty'.



Try now: chatma.ps

Source code available at: <https://github.com/ChatMaps/ChatMaps>

Thank you!



Demo

