AroundMe

Sprint 3 Planning Document

Team 12

Zuoyang Ding

Weichu Hu

Siyu Jiang

Wendi Zhang

Kenny Zheng

Sprint Overview

During Sprint 3, our goal is to finish all remaining tasks of the project backlog. All tasks from Sprint 1 and Sprint 2 are completed, with the exception of some UI designs that need to be modified. These modifications should be completed by the first week of Sprint 3. One important task for this sprint is to let users report inappropriate behaviors or harmful messages so that administrators can perform appropriate actions such as banning or deleting an account accordingly. The rest of our time will be spent to support error handling and enhancing the server to ensure the smooth of our app.

Scrum Master: Weichu Hu

Meeting Plan: Monday/Thursday/Friday LWSN Building (7:00pm - 10:00pm)

Risks and Challenges:

Due to the fact that all the intense works of our project are in sprint 2, in sprint 3, the user's stories are comparatively simpler than other two sprints. The main challenge we have for sprint 3 is to set up maintenance and ensure quality ethics among the users and within the group conversation. In this sprint, we will set up "report" buttons both within a group and user's profile page, which allows reporting of a user or a group when illegal or harmful messages were been sent. Any instance of reporting will be tracked and when a certain user or group were reported multiple times under same reason, actions such as banning a user or disbanding a group will be conducted by an administrator. In addition, by the end of sprint 3, we will have to re-module our UIs to achieve maximum appearance for the users, as well as develop test cases that verify each of our user's stories work properly.

Update Sprint 2 Tasks:

(User stories 4, 6, 11 are from previous sprint and need more UI enhancement as discussed.)

User Story #4:
As a user, I want to be able to join a group.

#	Description	Time	Owner
1	Module UI design (edit)	5hr	Wendi Zhang
2	~~Fetch user's information and store it to the correct group class~~	~~5hr~~	~~Kenny Zheng~~
3	~~Fetch correct group information and return it to frontend~~	~~5hr~~	~~Kenny Zheng~~

- Given that the UI is successfully implemented, when users want to view and join a group, they can access the group by pressing the group icon and join the group my pressing the add button inside a group
- ~~Given that the database is able to handle new members to the group when the user join the group, the database will be able to get the users information and store them accordingly~~
- ~~Given that the database is able to send information back to app when a group's information is updated, the database will update the display on app accordingly~~

User Story #6:
As a user, I want to be able to make a temporary group become permanent before it expires (when I think the topics and people in the group are interesting).

#	Description	Time	Owner
1	Module UI design (edit)	5hr	Wendi Zhang
2	~~Record the time after the group been created~~	~~5hr~~	~~Siyu Jiang~~
3	~~change the group type from temporary to permanent in a group class to prevent it from deleting after time session expired~~	~~4hr~~	~~Kenny Zheng~~

Acceptance Criteria:

- Given that the UI is successfully implemented, when users click on the button at the top-right corner, they should be able to access a drop down menu with a countdown timer, and a button to vote for a group to become permanent.
- ~~Given that the database is able to store a countdown timer when a new group is created, the database will be able to set a time for it and start counting down.~~
- ~~Given that database is able to store group permanently, when a group is changed to permanent, the database will never automatically delete it~~

User Story #11:
As a user, I want to check history messages in a group within 24 hours. (chat function + time count function)

#	Description	Time	Owner
1	Module UI design	5hrs	Wendi Zhang
2	~~Modify group data structure to contain messages~~	~~10 hrs~~	~~Zuoyang Ding + Kenny Zheng~~
3	~~Implement function that store and fetch messages in device and firebase accordingly~~	~~10 hrs~~	~~Zuoyang Ding + Kenny Zheng~~

Acceptance Criteria:
- Given that UI is successfully implemented, the users should be able to scroll through messages fluently.
- ~~Given that group data struct is correctly implemented, a group should be able to contain messages within it~~
- ~~Given that fetch function is correctly implemented, the user should be able to get messages they want~~

Sprint 3 Tasks:

User Story #18:
As a user, I want to be able to report particular users with inappropriate activities.

#	Description	Time	Owner
1	Module UI design	4 hrs	Wendi Zhang
2	Report a user with inappropriate activity or harmful messaging	7 hrs	Kenny Zheng

#	Description	Time	Owner
3	Provide feedbacks or reasoning for the report	7 hrs	Kenny Zheng

Acceptance Criteria:
- Given that UI is implemented correctly, the user should be able to use report buttons to file a report.
- Given that the function is implemented correctly, a user can report other users for inappropriate activities and harmful messaging.
- Given that the UI is implemented correctly, each report submitted by a user will be recorded in database along with reasoning.

User Story #19:
As a administrator, I can delete or ban a user who have inappropriate activities and reported by other users.

#	Description	Time	Owner
1	Modify UI to allow reporting of a user	5 hrs	Wendi Zhang
2	Keep track of reporting instance by other users	8 hrs	Kenny Zheng
3	Delete or ban a user accordingly to the reports	8 hrs	Kenny Zheng

Acceptance Criteria:
- Given that the UI is successfully implemented, a user can report other users for inappropriate activities or messages.
- Given that the database is implemented correctly, each report instance will be recorded in database as well as report counts
- Given that the report counts reached certain amount, the administrator can either delete or ban a user accordingly to the reports

User Story #20:
As a administrator I can delete or ban a group which used for discuss illegal information.

#	Description	Time	Owner
1	Modify UI to allow reporting of a group	6 hrs	Wendi Zhang
2	Keep track of reports sent by the users	8 hrs	Weichu Hu

#	Description	Time	Owner
3	Perform appropriate actions such as ban or delete a group accordingly to the reports	8 hrs	Weichu Hu

Acceptance Criteria:
- Given that the UI is successfully implemented, a user can report a group for inappropriate activities or conversation.
- Given that the database is implemented correctly, each report instance will be recorded in database as well as report counts
- Given that the report counts reached certain amount, the administrator can either delete or ban a group accordingly to the reports

User Story #21:
~~As a user I want to find roommate by this app (if time allows).~~ (**update: we will not do this task as time limited**)

User Story #22:
As a user I want the Android application to be responsive to all user requests

#	Description	Time	Owner
1	Module UI design	2 hrs	Wendi Zhang
2	The app will handle all user requests and minimize the chance of crashing	6 hrs	Zuoyang Ding
3	The app will prevent crashing by display messages	6 hrs	Zuoyang Ding

Acceptance Criteria:
- Given that the UI is correctly implemented, the client should be able to handle all user actions without crushing.
- Given that the function is implemented correctly, the app will be able to handle all user requests instead of crashing the app
- Given that the function is implemented correctly, the app will prevent crashing by display error messages

User Story #23:

As a user I want both the client and server to gracefully handle errors and the failure of other components that they interact with.

#	Description	Time	Owner
1	Prevent crashing of the app by display an error message	10 hrs	Zuoyang Ding
2	Error checking for every position crashing edges	8 hrs	Zuoyang Ding
3	Minimize the chance of crashing of the app	2 hrs	Kenny Zheng

Acceptance Criteria:
- Giving that the function and UI are implemented correctly, the app will display an error message to prevent the app from crashing
- Given that the function is implemented correctly, the server should be able to handle all requests and provide reasonable solutions/outcomes instead of crashing the app
- Given that the functions and the structures of the app are implemented correctly, the app should jump to login page instead of app crashing

User Story #24:

As a user I want the server design to scale to support additional load without requiring any change to the API or client code.

#	Description	Time	Owner
1	Improve image compression efficiency with image upload functions.	7 hrs	Weichu Hu
2	Improve image decode efficiency with image load display functions.	7 hrs	Weichu Hu

Acceptance Criteria:
- Update setting with bitmap function which called when upload image to firebase.
- Update setting with decode function which called when decode and display image in chat room.

User Story #26:
As a user, I want my temporary group disappear after it expired

#	Description	Time	Owner
1	Scanning all group for expired groups	8 hrs	Siyu Jiang
2	Implement backend function to smoothly remove groups from database without crash	8 hrs	Siyu Jiang
3	Implement front end function that represent the change in database	8 hrs	Siyu Jiang

Acceptance Criteria:
- Given that the group timestamp is counted correctly, the expired groups always be removed
- Given that the database successfully remove the group, the backend does not crash when a group is removed
- Given that the backend is correctly implemented, the frontend does not crash when a group is removed, and the UI updates on time

Testing Method:

We will be using automated unit test for all our testings using Espresso provide by Android Studio. By using Espresso testing, we will be able to test each user stories and functions individually to ensure each user's stories and functions work as expected. In addition, by using Espresso, we are also able to test all the buttons, text fields and other required fields in an UI page to ensure each field perform as expected without crashing or malfunction, as well as to ensure the flow of the UI smoothly.

Test Tasks:

Test case 01 Report user with inappropriate activity
> Severity: 2
> Instruction:
> 1. Login with a valid email and password
> 2. Join in a group and report a user
> 3. Check the database to see if the user has been marked
>
> Expected result:
> > In database, this user ID should be in a Class named ReportedUser

Test case 02 Administrator (Develops right now) ban user
> Severity: 2
> Instruction:
> 1. Administrator login into the Firebase
> 2. Use a user ID to find the the message that the user sended in ChatMessage Class
> 3. Check the message that the user sended
> 4. Disable the user in Authentication in Firebase
>
> Expectation:
> > The banned user cannot login by his/her valid email and password

Test case 03 Administrator (Develops right now) ban group
> Severity: 2
> Instruction:
> 1. Administrator login into Firebase
> 2. Use a group ID in ReportedGroup class to find the messages belong to this group
> 3. Check the messages in ChatMessage class
> 4. Change the value ban in Group class to true
>
> Expectation:
> > When user joined this banned group, cannot see the history message and cannot sent message

Test case 4: As a user I want the Android application to be responsive to all user requests

Severity: 2
Instruction:
  1. Running app on multiple devices.
  2. Let multiple device accessing different function and data at same time

Expected result:
The app should function normally, modification in data should also reveal on other device.

Test case 5: As a administrator, I can delete or ban a user who have inappropriate activities and reported by other users.
  Severity: 2
  Instruction:
  1. Create and new user and add it to a group
  2. Report that user multiple times

  Expected result:
  When the user login his/her account again, he will be notified that he cannot using the app for certain amount of time.

Test case 6: As a user I want the server design to scale to support additional load without requiring any change to the API or client code.
  Severity: 1
  Instruction:
  1. Create many groups
  2. Create many new users
  3. Associate all new users to all new groups
  4. Running the app

  Expected result:
  The app should run fairly smoothly, and don't crash.

Test case 7: As a user, I want my temporary group disappear after it expired
  Instruction:
  1. Create a group
  2. Wait until it expired

  Expected result:
  The group should be removed from database, and not showed on UI any more.

Remaining Backlog (23/24 Stories Completed)

1. ~~As a user, create an account through my current Google account~~
2. ~~As a user, I want to be able to use my favorite nickname (which can be the same as others)~~
3. ~~As a user, I want to be able to see the groups around me.~~
4. ~~As a user, I want to be able to join a group.~~
5. ~~As a user, I want to be able to create a group with topic tags I like.~~
6. ~~As a user, I want to be able to make a temporary group becomes permanent before it expires (when I think the topics and people in group are interesting)~~
7. ~~As a user, I want to be able to be removed from a group.~~
8. ~~As a user, I want to be able to upload and change my avatar as I want.~~
9. ~~As a user, I want to able to sort the group by activeness.~~
10. ~~As a user, I want to know when the group will be disbanded.~~
11. ~~As a user, I want to check history messages in a group within 24 hours.~~
12. ~~As a user, I want to sort the groups by tags.~~
13. ~~As a user, I want to be able to vote to decide whether a group will become permanent.~~
14. ~~As a user, I want to be able to send pictures in group chats.~~
15. ~~As a user, I want to be able to modify my personal profile.~~
16. ~~As a user, I want to be able to see profile information of others.~~
17. ~~As a user, I want to be able to protect my personal profile and only display with certain users.~~
18. ~~As a user, I want to be able to report particular users with inappropriate activities.~~
19. ~~As an administrator, I can delete or ban a user who have inappropriate activities or reported by other users.~~
20. ~~As an administrator, I can delete or ban a group which used for discussing illegal information.~~
21. As a user, I want to find a roommate by this app (if time allows).
22. ~~As a user, I want the Android application to be responsive to all user requests.~~

23. As a user, I want both the client and server to gracefully handle errors and the failure of other components that they interact with.

24. As a user, I want the server design to scale to support additional load without requiring any change to the API or client code.

25. As a user, I want to be able to log in with my own email and password

26. As a user, I want my temporary group disappear after it expired