



# AroundMe

## Sprint 1 Planning Document

### **Team 12**

Zuoyang Ding

Weichu Hu

Siyu Jiang

Wendi Zhang

Kenny Zheng

## **Sprint Overview**

During the first sprint, our main concern is to implement Firebase into our project and make sure it's functional and will be able to handle requests. Members of the team will not be divided into frontend or backend team, rather, team members will be assigned tasks based off of specialties and field of comfortable. At a maximum of two members of the team will be assigned to work on the frontend, they will be responsible for designing and creating the user interface. Rest of team members will be focusing on the backend, where they will be implementing controllers for the user interface, as well as database logic that can be used to handle any user requests.

**Scrum Master:** Weichu Hu

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

### **Risks and Challenges:**

There are significant amount of work need to be done for this sprint as well as many challenges, therefore it will be easy for the team to fall behind which might put us at a disadvantage for later sprints. One of the challenges our team faces is being able to implement Firebase and make sure it is functional. Since our team did not have many experiences on Firebase, therefore, many hours of sprint one will be spent on watching tutorials and experiencing on how to implement a functional database. Another challenge our team faces is due to the recent update of Android Studio, frontend team members will have to be familiar with Android Studio, and be able to create a clean and suitable UI for the user.

## Current Sprint Detail

User Story #1:

As a user, I would like to create an account through my current Google account.

#	Description	Time	Owner
1	Module UI design	5hr	Wendi Zhang
2	Import Google login authorization API	6hr	Zuoyang Ding
3	Store user account information into database	6hr	Kenny Zheng
4	Store user information without image and group ID, set Logout function for user	6hr	Zuoyang Ding

Acceptance Criteria:

- Given the UI is successfully implemented, when users open the app, they should be able to see login page and directed to correct page by pressing button
- Given the Google API is successfully imported, when users try to log in, they should be directed to Google authentic page and verified by Google.
- Given the database is successfully imported, users' account information should be able to be stored in the database, and the application should be able to recognize the user every time they log in.

User Story #2:

As a user, I want to be able to use my favorite nickname (which can be the same as others).

#	Description	Time	Owner
1	Module UI design	5hr	Wendi Zhang
2	Fetch user's nickname from frontend and store the nick into the correct user class	5hr	Weichu Hu
3	Fetch user's nickname from database and return to frontend	5hr	Kenny Zheng

Acceptance Criteria:

- Given that the UI is successfully implemented, when users want to change the nickname, they should be able to access their profile by pressing a setting button and change their nicknames.
- Given that the database is able to accept the request of changing nickname when users change their nickname, the database will get the new nickname and store in the correct user class.
- Given that database is able to synchronize updated user information to all related users, when a change is made in user class, the database will update the information to all related users.

User Story #3:

As a user, I want to be able to see the groups around me.

#	Description	Time	Owner
1	Module UI design	5hr	Wendi Zhang
2	Import Google Map API, and also apply Android device's GPS to locate device's position	7hr	Kenny Zheng
3	Fetch appropriate group location data from database and return to frontend	5hr	Zuoyang Ding

Acceptance Criteria:

- Given the UI is successfully implemented, the users should be able to see a list of groups around them when they are on the home page.
- Given that the Google API is successfully imported, maps should be shown in the application.
- Given that the database is able to send group location to the app, users should be able to use the function involving the usage of location information

User Story #4:

As a user, I want to be able to join a group.

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

Acceptance Criteria:

- 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 by pressing the add button inside a group
- Given that the database is able to handle new members to the group when the user joins the group, the database will be able to get the user's 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 #5:

As a user, I want to be able to create a group with topic tags I like.

#	Description	Time	Owner
1	Module UI design	5hr	Wendi Zhang
2	Fetch topics from frontend and create a new group class in database	7hr	Zuoyang Ding
3	Fetch the group information just created and return it to frontend	5hr	Siyu Jiang

#### Acceptance Criteria:

- Given the UI is successfully implemented, the users should be directed when they click on the button to a page where they can edit group name, information, and tags.
- Given that database is able to store tags, when a group is labeled with a certain tag, the database will store it into the group's class.
- Given that the database is able to send information back to app when a group's information is updated, the database will update the app display 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	5hr	Weichu Hu
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 #7:

As a user, I want to be able to exit from a group.

#	Description	Time	Owner
1	Module UI design	5hr	Weichu Hu
2	Fetch the user's information from frontend	1hr	Kenny Zheng
3	Find the correct group class and delete the user information in that group class (update group information in database)	7hr	Siyu Jiang

Acceptance Criteria:

- Given that the UI is successfully implemented, when users click on a button at the top-right corner, they should be able to access a drop down menu where they can choose to leave a group.
- Given that the database is able to store a list of users in each group when a user chooses to exit a group, the database will remove his/her user ID from the list.
- Given that the user exits a temporary group successfully, the group will no longer be shown in the list of groups

User Story #8:

As a user, I want to be able to upload and change my avatar as I want.

#	Description	Time	Owner
1	Module UI design	5hr	Weichu Hu
2	Fetch user information and new avatar from frontend	3hr	Kenny Zheng
3	Find the correct user class and update the user's avatar in this user class	5hr	Siyu Jiang

	(update the particular user class in database)		
--	--	--	--

#### Acceptance Criteria:

- Given that the UI is successfully implemented, when users tap on the profile picture, they can access profile page and edit information as needed.
- Given that the changing avatar function is successfully implemented, when user want to change their avatar, they should be able to upload picture from local device
- Given database can store and update avatar successfully when users change their avatars, the database will update and store the avatar in the user class.
- Given that the user successfully changed their avatar, other users should be able to see the new changes as well as change the avatar and necessary information in the database

#### Remaining Backlog (8/24 Stories Complete)

- ~~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.