

# ***Team 17 - WeGroup***

## **Product Backlog**

Hanqi Du, Ao Liu, Xiaoyang Lin, Wanqi Jiang, Qinxin Shi, Zhixuan Yu

### **Problem Statement**

Nowadays group-message applications play an important role in everyone's daily life. A problem arises where a group chat with too many people can get messy and unorganized. It takes time to read through all the messages but the fact is not all of them matter. The idea of our app is to allow users to have a more organized group chat by creating branches (or "topics") inside the conversation, so that it's easier for users to find what people talked about on the topics that they are interested in.

### **Background Information**

This is a problem that came to many of us that in our messenger apps, many group conversations can get very messy. We open the app, and there are 200 unread messages. We would have to read through all of them to find the important information. Our app allows users to hashtag topics so that it is easier to organize conversations, and find out what people have been talking about.

### **Environment**

For the front end, we will use mainly Swift, and Xcode to implement with Cocoa library.  
For the database, we will use SQLite to manage and update the user data.

### **Requirements**

Backlog ID	Functional Requirements	Hours
1	As a user, I would like to register an account	15
2	As a user, I would like to create my own profile	8
3	As a user, I would like to retrieve password when I forget (if time allows)	8
4	As a user, I would like to add friends	6
5	As a user, I would like to delete friends	6
6	As a user, I would like to create new direct conversations	15
7	As a user, I would like to create group conversations	15

8	As a user, I would like to search a friend in contacts	8
9	As a user, I would like to search a keyword in conversations (if time allows)	15
10	As a user, I would like to view groups that a friend is in	15
11	As a user, I would like to create topics within a group chat	20
12	As a user, I would like to view messages sorted by topics	30
13	As a user, I would like to get notifications of the topics I am interested in	16
14	As a user, I would like to mute messages under topics I am not interested in	8
15	As a user, I would like to invite friends to groups	15
16	As a user, I would like to mute direct and group conversations	15
Total		161

Backlog ID	Nonfunctional Requirements
1	We want the users to run the app on iOS devices
2	We want to make sure that the UI for each of our applications is modern and aligns with current best practices on it's respective platform
3	We want to keep important information like password, user profile information and chatting history secure and private
4	We would like users to add many friends/have many group conversation without causing any problem to the app or server
5	We want the response time of adding friends and searching to be under 2 seconds
6	We want to make sure that our service can handle multiple users, but being able to serve millions of people is not currently our priority

**Use cases:**

Case 1: Register an account	System responses
1.Click on the “register” button	2. The register form appears
3. Filling the form	
4. Click the “submit” button	5. The form dismiss

Case 2: Create my own profile	System responses
1. Open the profile setting	2. The profile form appears
3. Filling in the form	
4. Click the “submit” button	5. The form dismiss

Case 3: Retrieve password when forget	System responses
1. Click on the “forgot my password” button	2.Dialog appears to ask for user’s email address
3. Input the email address the user used for registration	4. Dialog appears to inform that the password is sent
3. Click on the “OK” button	4. Dialog disappears

Case 4: Add a friend	System Response
1.Use Search button to search User ID	2.List Users that match
3.Select one user and click “ADD” button	4.Get permission from friend and add this friend to friend list

Case 5: Delete a friend	System Response
1.Select one friend from friend list	
2.Click delect button	3.Delete this friend from friend list

Case 6: create new direct	System Response
1. Click a friend to make conversation	2. Build dialog box for 2 users

Case 7: Create group conversations	System Response
1. Click “create group conversation” button	2. List all the contacts
3. Choose more than 2 friends	4. Group conversation appears

Case 8: Search a friend in contacts	System Response
1. Select the contacts	2. Search bar dialogue appears
3. Type the keyword	4. List all the contacts contain the keyword

Case 9: Search a keyword in conversations (if time allow)	System Response
1. Choose the search bar	2. Search bar dialogue appears
3. Search the keyword	4. List the conversations that contain the keyword

Case 10: View groups that a friend is in	System Response
User taps the friend profile	The user profile appears
User will see a section of groups this friend is in	Profile shows a section of groups
User taps the section of groups	A list of groups appears

Case 11: Create topics within a group chat	System Response
User taps ‘create topic’ button	System shows a dialog asking for topic name
User press ‘enter’ on keyboard	A new topic with given name is created

Case 12: View messages sorted by topics	System Response
User taps a specific topic	A sorted list of message under this topic appears

Case 13: Get notifications of the topics the user is interested in	System Response
The user goes into a group chat, and clicks on a topic go view messages under the topic, and then click on the “interested” button to follow the topic.	A small view appears for about a second to indicate that the user has successfully followed the topic.
After following a topic, next time a message under that topic is received, the user will get a notification.	The system sends a notification to the user that he/she has received a new message.
The user adds a new topic.	The system sets the user to follow the topics that he/she created by default

Case 14: Do not get notifications of messages under topics the user is not interested in	System Response
The user does not click the the “interest” button when viewing a topic	The system will not send notifications to the users when messages under the uninterested topics are received

Case 15: Invite friends to group conversations	System Response
The user creates or opens a group conversation, opens up the “info” panel of the group, and clicks the “add” button.	The system pops up a view allowing user to search friends by user_id or user_name
The user selects a friend.	The system adds the friend to this group conversation.

Case 16: Mute direct and group conversations	System Response
The user opens up a direct or group conversation, opens up the “info” panel of the conversation, and	The system will not push notification to the user when the user receives messages from the muted

clicks the “mute” button.	conversations.
---------------------------	----------------