CS 408 Team 10

Kenny Tam

Milia Enane

Michael Rollberg

Jake Losin

Brandon Nice

Phase 2 Test Plan

Backlog ID 1: As a user, I would like to register an account for WeGroup

- Functionality: User presses the signin button and inputs a username, email, and
  consistent passwords
    - Result: Account created dialog appears, user can continue to use the app

      Critical test case

- Equivalence: User enters username that is already taken in the username field
    - Result: The user should be prompted that the username has already been taken
      and no new account should be created

      Critical test case

- Boundary: User enters more than 14 characters in the username field
    - Result: A popup should appear notifying the user that they should only have up to
      14 characters allowed

      Important test case

Backlog ID 2: As a user, I would like to log into WeGroup with username and password

- Functionality: User enters correct username and password
    - Result: Application acknowledges the user, allows them to use the app

Critical test case

● <u>Equivalence</u>: User enters correct username but wrong password

  - Result: App tells the user to enter their password again

  Critical test case

● <u>Equivalence</u>: User enters username that has not been registered yet

  - Result: User is prompted to re-enter their username

  Critical test case

● <u>Boundary</u>: User inputs an arbitrarily large amount of input for a username

  - Result: App should prevent user from inputting enough characters to crash the

    app

  Critical test case

Backlog ID 3: As a user, I would like to create my own profile (region, bio)

● <u>Functionality</u>: The user navigates to their profile page, enters text for region and bio,

  selects the "upload" button

  - Result: The user's region and bio information is updated and can be seen by

    other users

  Important test case

● <u>Equivalence</u>: The user navigates to their profile page, enters text for region and bio, and

  repeats the process a second time with different text inputs for both fields

  - Result: The user's region and bio information is updated to the newly create input

  Important test case

● <u>Boundary</u>: User enters nothing into the region and bio fields

  - Result: The app should allow for blank information

Workaround test case

- <u>Boundary</u>: The user inputs an arbitrarily large number of characters into the profile bio
    - Result: The user will be prompted to enter a set amount of characters or display all of the contents put into a user's bio

    Important test case

Backlog ID 4: As a user, I would like to update and change my profile picture

- <u>Functionality</u>: User taps on their picture in their profile section and an option appears to upload a photo from the photo library.
    - Result: User uploads a photo and sets it as their profile picture.

    Important test case

- <u>Functionality</u>: User taps on their picture in their profile section and an option appears to take a photo via camera.
    - Result: User takes a photo and sets it as their profile picture.

    Important test case

- <u>Equivalence</u> : User taps on their picture in their profile, uploads a picture from their photo library, and repeats the process a second time
    - Result: The first photo is replaced with the second photo

    Important test case

- <u>Boundary</u>: User selects their picture and zooms it in all the way
    - Result: The photo is cropped to fit the circle in the profile view

    Workaround test case

Backlog ID 6: As a user, I would like to add contacts by typing their usernames

- <u>Functionality</u>: User selects the "add friend" button and types the correct username of their desired friend
    - Result: The friend is added to their contacts and is displayed on the "contacts" page

        Critical test case

- <u>Equivalence</u>: User enters their own username
    - Result: The user cannot their own username to their list of contacts

        Workaround test case

- <u>Equivalence</u>: User types a non-existing username and tries to add that user as a contact
    - Result: The app should tell the user that a user with that username does not exist

        Important test case

- <u>Boundary</u>: User enters a username that is longer than 14 characters (no accounts should have such username because the character limit is 0-14)
    - Result: The app tells the user that the entered username is too long

        Important test case

Backlog ID 7: As a user, I would like to delete contacts

- <u>Functionality</u>: User swipes left on a contact in their contacts list and presses the delete button
    - Result: The contact is removed from the list

        Critical test case

- <u>Equivalence</u>: The user adds a friend, deletes them, and adds them again

- Result: The app should add the friend to the user's contact list, remove it, and add it back again.

   Critical test case

● Boundary: User goes through their entire contact list and deletes everyone

- Result: The app should properly remove every contact

   Critical test case

Backlog ID 8: As a user, I would like to create new direct conversations

● Functionality: User navigates to the "chat" tab, selects the plus button and selects desired friend to chat with.

- Result: A conversation opens up and the two users can message each other directly

   Critical test case

● Equivalence: User selects the plus button, selects a user with whom they already have a conversation with

- Result: App alerts user that there is an existing conversation with that friend

   Important test case

● Boundary: User selects multiple friends for a conversation

- Result: App creates a multi-user conversation

   Critical test case

Backlog ID 9: As a user, I would like to send messages in a direct conversation

● Functionality: The user starts a new conversation, selects the new user (receiver), types in a message in the chat bar, and hits send.

- Result: The receiver and the user should see the message sent with the correct corresponding user id and profile picture

  Critical test case

- Equivalence: The user opens a pre-existing conversation from the menu that already has messages and sends a new message
  - Result: The user and the receiver both see the message with correct corresponding user id and profile picture

    Critical test case

- Boundary: User sends a message to multiple friends in a conversation
  - Result: The user and everyone in the conversation can see the message with the corresponding ids and profile pictures attached to who sent it

    Critical test case

Backlog ID 10: As a user, I would like to receive messages sent from users in my contact list

- Functionality: A user starts a conversation with a second user and the second user types a message
  - Result: The first user will see the message sent from the corresponding user

    Critical test case

- Equivalence: A user creates a chat with multiple other users and one each one of the other users sends a message
  - Result: User should see the messages and the senders of the message in the other that they are sent

    Critical test case

- <u>Boundary</u>: User receives a message from another user, but then deletes the other user from their contacts list
    - Result: The user should still be able to see the message, but not the name of the sender

    Important test case

Backlog ID 11: As a user, I would like to create group conversations

- <u>Functionality</u>: User creates a new chat and selects multiple contacts to be included in the chat
    - Result: The conversation should be visible in the chat page

    Important test case

- <u>Equivalence</u>: User adds multiple contacts to a chat, but the other contacts don't have each other in their contact lists
    - Result: All users in the conversation will see each other's messages and the other users should be added to each other's contact lists

    Workaround test case

- <u>Boundary</u>: User creates a conversation with no users
    - Result: User is prevented from being able to do so and needs a minimum of one contact (although that would itself still not be a group conversation)

    Important test case

Backlog ID 12: As a user, I would like to send messages in a group conversation

- <u>Functionality</u>: User selects a group conversation and sends a message to the group

- Result: Other users in the group conversation receive the message

  Critical test case

- Equivalence: User sends message to group, but hasn't added all members to contacts

  - Result: The previously unadded contacts should now be in the user's contact list

  Critical test case

- Boundary: User sends message with no text

  - Result: App should tell user to enter non-blank text or prohibit the message from being sent

  Workaround test case

Backlog ID 13: As a user, I would like to receive a message from the group conversations I am in

- Functionality: Another user in a group chat sends the group a message

  - Result: The current user receives the group message as does everyone else in the group

  Critical test case

- Equivalence: Every member of the group sends a message to the group

  - Result: Every member of the group can send and receive all messages with no discrepancies or errors in profile images or names

  Important test case

- Boundary: User receives a group message with no text

  - Result: App should not allow the sender to send blank group messages

  Workaround test case

Backlog ID 14: As a user, I would like people in the groups I'm involved in to be added to my contacts automatically

- Functionality: A user is put into a group with people that are not in their contacts
    - Result: The users that are not in the current user's contacts are added to their contacts

        Important test case

- Equivalence: A user is put into a group with people not in their contacts list and deletes the newly added contacts
    - Result: The contacts should be re-added if the user interacts with/is interacted with by the group again as it is necessary to have the contacts in the list to receive messages

        Important test case

- Boundary: All the users in the group are already in the contact list
    - Result: The app recognizes this and does not add any further users to the contact list

        Workaround test case

Backlog ID 15: As a user, I would like to delete conversations

- Functionality: A user swipes left on a conversation box and presses delete
    - Result: The conversation is removed from their chat wall

        Important test case

- Functionality: User logs out of the application
    - Result: All conversations are deleted

        Critical test case

- Functionality: User swipes left on a conversation box but then swipes right to close the delete option

  - Result: The conversation is not removed

    Workaround test case

- Equivalence: User swipes left on a conversation box to reveal the delete option and then swipes right on it

  - Result: The delete option is removed and the conversation box appears as it did before

  - **Workaround test case**

- Boundary: User deletes all of the existing conversations

  - Result: The app should allow for this, and delete all conversations

    Important test case

Backlog ID 16: As a user, I would like to search a friend in contacts

- Functionality: The user presses the search box on and types in characters

  - Result: The contact wall should only show contacts whose username starts with those characters

    Important test case

- Functionality: The user types characters in the search, but then presses cancel

  - Result: The full contact list should appear, disregarding the characters inputted into the search. Also the characters should be removed from the search bar

    Workaround test case

- <u>Boundary</u>: The user presses the search box, but does not type any characters, then hits cancel
  - Result: The app should show the initial contact list without any contact being filtered out

  Workaround test case

Backlog ID 18: As a user, I would like to search for groups that a friend is in

- <u>Functionality</u>: The user presses the search box and enters characters
  - Result: The app shows all of the chat conversations that the desired user is in

  Important test case

- <u>Functionality</u>: The user types in the search, but then presses cancel
  - Result: The chat wall resets as if nothing was searched for, and the characters in the search box should be cleared

  Workaround test case

- <u>Equivalence</u>: The user presses the search box and enters non-alpha-numeric characters (emojis, punctuation, etc.)
  - Result: The search is able to identify users with special characters in their name

- <u>Boundary</u>: The user types in the search, does not enter anything, then presses cancel
  - Result: The app shows the full, unaltered chat list

  Workaround test case