

Logan Reid

Grant Muslow

Kristian Thomas

Ryan Cunningham

Project Requirements as a Collection of User Stories

User Story 1: User opens the app for the first time.

User opens the app. User taps the “Don’t have an account? Register” button. User types their username. User types their email. User types their password. User confirms their password. User taps “Register”.

Test Case:

Inputs:

App is accessed

Account = false

Input(username,email,password)

confirmPassword = true

ExpectedOutput = account

ActualOutput = account

Pass/Fail Criterion = ExpectedOutput = ActualOutput

User Story 2: User wants to message a friend for the first time.

User taps the pencil (create chat button). User types the friend’s name. User taps on friend’s name. User taps “Start Chatting”. User types their message. User presses send.

Test Case:

Inputs:

Chat: accessed

Input(friendName)

AccountName: accessed

Input(message)

MessageSent = true

Delivered = true

Expected Output: MessageSent

Actual Output:

Pass/Fail Criterion: MessageSent = true

User Story 3: User wants to message their friend (chat already created).

User taps the “Chats” button. User presses the friend’s name. User types their message. User presses send.

Test Case:

Chat: accessed

Friend: accessed

Input(message)

MessageSent = true

Delivered = true

Expected Output: MessageSent & Delivered = true

Actual Output:

Pass/Fail Criterion: Delivered = true

User Story 4: User wants to message their friend, then another friend.

User taps the “Chats” button. User presses the friend’s name. User types their message. User presses send. User presses the back button. User presses the second friend’s name. User types their message. User presses send.

Test Case:

Chat: accessed

Friend: accessed

Input(message)

MessageSent = true

Delivered = true

Chat = accessed

Friend2: accessed

Input(message)

MessageSent = true

Delivered = true

Expected Output: MessageSent & Delivered = true

Actual Output:

Pass/Fail Criterion: Delivered = true

User Story 5: User wants to delete a message thread.

Users taps the “Chats” button. User holds down the chat they want to delete. User presses delete. The chat disappears.

Test Case:

Chat: accessed

HoldButton: true

OptionsMenu: on

Delete = true

Chat: removed

Expected Output: Chat = removed

Actual Output:

Pass/Fail Criterion: Chat = removed

User Story 6: User wants to create a group chat.

User taps the pencil (create chat button). User selects an image for the group. User types a name for the group. User types a description for the group. User types friend's name. User taps on friend's name. User repeats previous two steps until the desired group is complete. User presses "create". User types their message. User presses send.

Test Case:

New Chat Settings = accessed

Input(image)

Input(chat name)

Input(description)

Input(friend's name)

Friend: accessed

(Repeat last 2 steps)

Create = true

Input(message)

MessageSent = true

Delivered = true

Expected Output: MessageSent & Delivered = true

Actual Output:

Pass/Fail Criterion: Delivered = true

User Story 7: User wants to join an existing group chat.

User taps the “Explore” button. User looks through list of existing groups. User presses on the group they’d wish to join. Group info page pops up. User presses join group. User types their message. User presses send.

Test Case:

Explore = accessed

DisplayGroups: on

Group: accessed

DisplayInfo: on

JoinGroup = true

Input(message)

MessageSent = true

Delivered = true

Expected Output: MessageSent & Delivered & JoinGroup = true

Actual Output:

Pass/Fail Criterion: Delivered & JoinGroup = true

User Story 8: User wants to edit their profile.

User taps the “Profile” button. User presses the pencil (edit button) next to their profile picture. User finds the part of their profile to edit. User taps the information to edit. User adds new information. User presses save.

Test Case:

Profile = accessed

Edit = on

Input(information)

Save = true

Expected Output: information & save = true

Actual Output:

Pass/Fail Criterion: information & save = true

User Story 9: User wants to see another user's created groups

User taps the "Explore" button. User taps the search box. User types in a user's username. User presses the desired username. User scrolls down to reveal that user's created groups.

Test Case:

Explore = accessed

SearchBox = accessed

Input(username)

Username = accessed

DisplayUserGroups = on

Expected Output: DisplayUserGroups = on

Actual Output:

Pass/Fail Criterion: DisplayUserGroups = on