

FLOCKR: NEW FEATURES AND THEIR PROSPECTS

November 05, 2020

PREPARED BY

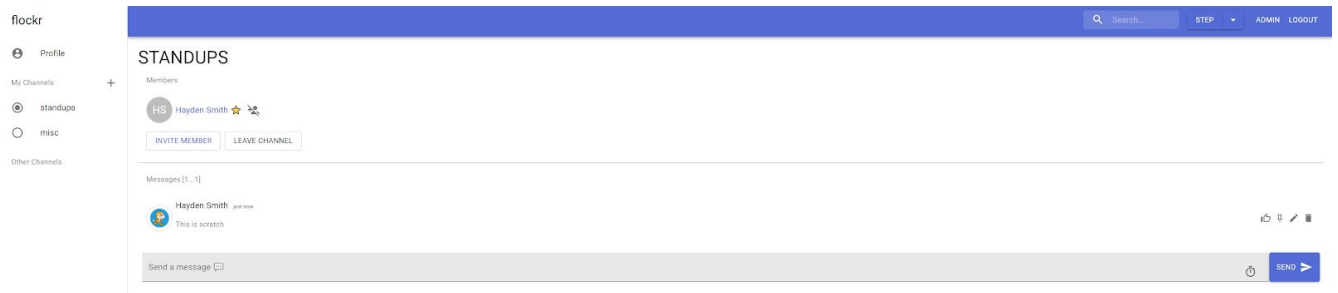
Wed15GrapeTeam2

Members:

- Chitrakshi Gosain
- Jordan Hunyh
- Cyrus Wilkie
- Joseph Knox
- Ahmet Karatas

Flockr Overview

Flockr is inspired by **Flock**, a team communication app and online collaboration platform, which comes with team messaging, project management and other great features.



UNSW was looking for a in-house digital collaboration and communication tool for groups and teams to support the high intensity learning environment that the transition from trimesters to hexamesters in 2020 would bring.

Therefore, rather than reinventing the wheel, UNSW found that **Flock** already has all the features it needs. Hence, it decided to build a new product named **Flockr** to connect with the younger and more “hip” generation that fell in love with flickr, Tumblr, etc.

UNSW has contracted out Pineapple Pty Ltd (a small software business run by Hayden Smith, lecturer COMP1531) to build the new product.

Pineapple Pty Ltd has sub-contracted two software firms:

FRONTEND DEVELOPERS

The initial web-based GUI was done by two software developers, Sally and Bob (Catdog Pty Ltd)

BACKEND DEVELOPERS

The backend python server was done by Wed15GrapeTeam2 (in this case, here) - a team of talented misfits completing COMP1531 in 20T3.

Details

The features of Flockr implemented by the team as per UNSW's request are:

1. Ability to login, register if not registered, and log out
2. Ability to reset password if forgotten
3. Ability to see a list of channels
4. Ability to create a channel, join a channel, invite someone else to a channel, and leave a channel
5. Within a channel, ability to view all messages, view the members of the channel, and the details of the channel
6. Within a channel, ability to send a message now, or to send a message at a specified time in the future
7. Within a channel, ability to edit, remove, pin, unpin, react, or unreact to a message
8. Ability to view user anyone's user profile, and modify a user's own profile (name, email, handle, and profile photo)
9. Ability to search for messages based on a search string
10. Ability to modify a user's admin permissions: (MEMBER, OWNER)
11. Ability to begin a "standup", which is an X minute period where users can send messages that at the end of the period will automatically be collated and summarised to all users

[Requirements] Elicitation

4 people were interviewed as target users by the team, and their names and email address collected. Target users were the people who currently use a tool like Flockr(Discord, Slack, Teams, etc.), or intend to.

The target users were presented with a series of questions to understand what problems they might face / have with teamwork-driven communication tools that are currently unsolved by Flockr.

The recorded responses for all target users interviewed are as follows:

Name: Thomas Szatmary

Email: thomasszatmary@gmail.com

1. What do you currently use an app like Flockr (Discord, Slack, Teams, etc.) for?

A: I use Discord to communicate with friends to discuss how we will work as a team to win the games we play.

I also use teams for school assignments to discuss group tasks.

2. Why do you choose to use this app over other similar apps?

A: Discord is easy, reliable and convenient to use. I also use teams for the same reason, except it has better team sharing capabilities.

3. What are some outstanding features of the app that you use?

A: text chat channels, voice chat, video calling, screen sharing, direct messaging.

4. Are there any specific features you wish to add to these tools?

A: perhaps have some inbuilt minigames, to build up team comradery.

5: Have you encountered any issues or complaints with your preferred app?

A: Sometimes the audio distorts or cuts out but that's it.

Name: Sabena Bhadri

Email: sabena.bhadri@gmail.com

1. What do you currently use an app like Flockr (Discord, Slack, Teams, etc.) for?

A. Use discord for society meetings and teams for class tutes

2. Why do you choose to use this app over other similar apps?

A. Bc everyone else uses them hehe

3. What are some outstanding features of the app that you use?

A. No lag, instant messaging, sound and video qual, recording capability

4. Are there any specific features you wish to add to these tools?

A. Wish there were better graphics on discord and pictures like being able to set pics for your channels

5. Have you encountered any issues or complaints with your preferred app?

A. Would like less visible features on teams, feels like ur about to click the wrong thing all the time and be unmuted or sth

Name: Akshay Singh

Email: akshaythacmp@gmail.com

1. What do you currently use an app like Flockr (Discord, Slack, Teams, etc.) for?

A: I use Discord. I think it's a fun place to hangout and meet with like minded people. I also use it for talking while gaming because it's very consistent and ubiquitous in the community. Also, I love how I can choose to be anonymous whenever I want. Makes me feel less vulnerable and safe in any community.

2. Why do you choose to use this app over other similar apps?

A: It's just the first app of this kind that I got into. I am comfortable with the interface, it's widely used and I really can't think of anything or way that another app could serve me better with. Not to say that any of its competitors are bad in any way. I'm just used to Discord.

3. What are some outstanding features of the app that you use?

A: I really like the interface and the ability to be anonymous. Also, I love the concept of exclusive servers. Group calls with many people at once also come really handy many times. And direct messaging is great too.

4. Are there any specific features you wish to add to these tools?

A: Yes, add themes. I just really want to see Discord go crazy with the themes. I would also like to see some annotation features as well for video calls. I also wish

that we were able to share screens in direct messages.

5: Have you encountered any issues or complaints with your preferred app?

A: No, I don't have any issues or complaints with Discord. It's just the first app of this kind that I got into. I am comfortable with the interface, it's widely used and I really can't think of anything or way that another app could serve me better with. Not to say that any of its competitors are bad in any way. I'm just used to Discord.

Name: Jono Chau

Email: chaujono@gmail.com

1. What do you currently use an app like Flockr (Discord, Slack, Teams, etc.) for?

A: I use discord to communicate with my friends while playing games and also to talk with members of university groups I am a part of.

2. Why do you choose to use this app over other similar apps?

A: Because it has a nice interface and is easy to use, and is accessible on a variety of platforms.

3. What are some outstanding features of the app that you use?

A: Gif search, ability to add bots which provide additional functions such as music or polls, also the ability to communicate and use these features in group chats as well as one on one conversations and audio/video calling for co-op games.

4. Are there any specific features you wish to add to these tools?

A: Watch videos at the same time as other server members (watch party), just like music bots

5: Have you encountered any issues or complaints with your preferred app?

A: Problems with installation, server crashes

[Requirements] Analysis & Specification - Use Cases

Once the target users were interviewed, and their responses recorded, it was time to consolidate the information gathered.

The responses from the elicitation were then thoroughly inspected to identify the requirements. These requirements were then expressed as **User Stories**.

The user stories were recorded in the following format:

As a < type of user >, I want < some goal > so that < some reason >.

The user stories were then modified to accompany the **Acceptance Criteria** as notes so that a clear definition could be provided of when a story has been completed.

These user stories were well documented and are as follows:

RQ:1

As a logged-in user, I want to be able to apply themes to the chat in direct messaging and server channels so that the chats can look more engaging and classic.

Acceptance Criteria

- User should be able to direct message another user
- User should have access to various themes for the chat
- User should be able to switch between the possible chat themes for each chat individually
- An option is placed in chat settings of each individual direct chat which allows both the users to change chat theme

RQ:2

As a logged-in user, I want to be able to have video and audio calls with my team to easily communicate ideas and discuss projects.

Acceptance Criteria

- The app should be able to handle a potentially indefinite call duration
- Users should be able to easily join and leave calls
- Users start off muted and have video off, which they can turn on
- A limit can be set to how many people are in a call

RQ:3

As a logged-in user, I want to be able to share screens in direct messaging so that I can communicate my ideas visually to the person on the other end in order to express my idea effectively

Acceptance Criteria

- User should be able to video call one to one with another user
- User should be able to share screen and device audio during the video call
- An option is placed on the profile page of each user which allows a person to request to start a one-on-one video call with that user
- User should receive a pop-up with options to accept or decline an incoming video call when it is received
- When a call is initiated users should be presented with a call interface which allows them to end the call at any time
- The call interface shows the live videos of users on both ends, and has option at the top to enable screen sharing

RQ:4

As a logged-in user, I want to be able to send messages privately to other users so that I can get individual feedback on my ideas and how I do tasks.

Acceptance Criteria

- The only members who should be able to access these messages are the sender and recipient

RQ:5

As a logged-in user, I want to be able to upload and share snippets of my work with my team members to better collaborate on our shared projects.

Acceptance Criteria

- Users should be able to upload local files to a channel for other members to download
- These snippets should include images, text files or any other type of file
- Only members in the channel should be able to access these files
- There should be a maximum upload size of 50MB per file

RQ:6

As a logged-in user, I want to be able to have access to annotation features during video calls especially when screen sharing so that i can easily share ideas

Acceptance Criteria

- User should be able to video call in group
- User should be able to screen share during a video call
- User should be able to access annotation tools such as whiteboard, marker, highlighter, stamp, eraser etc.
- An option is placed on the profile page of each user which allows a person to request to start a one-on-one video call with that user
- User should receive a pop-up with options to accept or decline an incoming video call when it is received
- When a call is initiated users should be presented with a call interface which allows them to end the call at any time
- The call interface shows the live videos of all users, and has options at the top to enable screen sharing and hence enable annotation features in next step

RQ:7

As a channel member, I want to be able to upload and set an image file to be the profile image for a channel, to make it easier to identify a channel's purpose.

Acceptance Criteria

- User should be able to upload images from their local machine to set as the profile image of a channel
- The image should be clearly visible when viewing the contents of the channel
- Every member of the channel should be able to change the profile image at any time
- Only members of the channel should be able to change the profile image
- The user should be able to select the dimensions of the image at the time of upload

RQ:8

As a logged-in user, I want to be able to play some simple games to better build up the relationship within our teams and boost comradery.

Acceptance Criteria

- Games should be common to all users, preferably run within the teamwork-driven communication tool.

- There should be no extra requirements to play these games other than the software itself

RQ:9

As a logged-in user, I want to be able to make audio call to individuals so I can chat with another person while continuing to play co-op games at the same time.

Acceptance Criteria

- User should be able to audio chat with another user one on one
- User should be able to audio chat in the background while using another application
- An option is placed on the profile page of each user which allows a person to request to start a one-on-one audio call with that user
- User should receive a pop-up with options to accept or decline an incoming audio call when it is received
- When a call is initiated users should be presented with a call interface which allows them to end the call at any time

The user stories orienting towards a similar *problem* were accounted for, together, and a use case was generated for the same.

The use cases developed attempt to tell a story of a solution that satisfies the requirements elicited by the target users.

The various use cases developed are:

- Calling
- Channel Profile Image
- Direct Messaging

These are documented in written-recipe style as follows:

CALLING

Use Case: Calling

Goal in Context: Users can call(audio/video) in group or one-to-one, share screens during the call and use annotation features

Scope: Flockr interface & backend

Level: Primary task

Preconditions: All users in the call are in the channel, no call already active

Success End Condition: Call successfully started with all users able to share video, audio and share screens.

Failed End Condition: Users unable to share audio, video or screen

Primary Actor: All users within the channel who are in the call

Trigger: Any user within the channel clicks the call button

Main Success Scenario

Step 1. On the channel page, user clicks the call button

Step 2. Other user(s) join the call

Step 3. User(s) turns on audio, video and/or share screens and access annotation features

Step 4. Users are able to hear and see each other

Step 5. All users leave the call

Step 6. The call automatically ends

CHANNEL PROFILE IMAGE

Use Case: Set Channel Image

Goal in Context: Set a given image file as the “profile image” of a given channel, which is clearly visible when viewing that channel

Scope: Flockr interface & backend

Level: Primary task

Preconditions: Channel exists in the Flockr, user provides a valid image file

Success End Condition: The given channel’s original profile image file is replaced with the provided image file. If there was no prior profile image file, the given image is set as the profile image.

Failed End Condition: The given channel’s profile image is unchanged. If there is no profile image, the channel continues to have no profile image.

Primary Actor: User

Trigger: User clicks the “set channel image” button in the frontend.

Main Success Scenario

- Step 1. User is a member of a channel in the Flockr.
- Step 2. User clicks the “set channel image” button in the frontend.
- Step 3. User drags a valid image file into the “drag image file here” field.
- Step 4. User clicks on the “set image” button.
- Step 5. The channel’s profile image is replaced with the image contained in

the provided file.

Step 6. This profile image is clearly visible to any user who is a member of the channel.

DIRECT MESSAGING

Use Case: Direct messaging

Goal in Context: Sending a direct message to another user individually

Scope: Flockr interface and backend

Level: Primary task

Preconditions: Both users have registered accounts, at least one user is logged in

Success End Condition: A chat message has been created that can only be viewed by the sending user and the specified, individual recipient user

Failed End Condition: No message is sent or exists on Flockr that is visible to only the sender and specified recipient

Primary Actor: User

Trigger: User clicks the send direct message button on the profile of another user

Main Success Scenario

Step 1. Flockr displays the messaging interface with messages displayed on the main part of the screen and a text box present at the bottom, with the name of the recipient user occupying the title position.

Step 2. The user types in the intended message into the text box.

Step 3. The user clicks the send button located immediately next to the message text box.

Step 4. The message is sent to the Flockr backend.

Step 5. The message is displayed in the messages section of the messaging interface.

[Requirements] Validation

With the completed use case work, the target users were reached again to inquire as to which extent the use cases were adequately describing the problem they are asking to solve.

The target users were asked for their comments and the comments were then documented, they are as follows:

Akshay: Yes this is precisely what I would like to see in the app. This will definitely improve my experience on the app.

Thomas: yep the requirements look good to me and seem to be perfect for what I need

Sabena: great extent, this is exactly what I was thinking/wanting, the success scenario describes exactly the kind of interface I'd want

Jono: Yes, these requirements definitely cover the kinds of things I'd want the app to do, particularly with regard to direct messaging.

[Design] Interface Design

After establishing a good understanding and conceptualisation of the *problem* (described in requirements), a blueprint of the solution was designed in terms of what capabilities would be necessary.

The capabilities are specified as the HTTP endpoints.

An interface table describing the necessary capabilities of the potential solution for each use case is listed below:

Function Name	HTTP Method	Parameters	Return Type	Exceptions	Description
channel/calling/poststream	POST	(token, channel_id, calling_buffer)	{is_success, call_type}	InputError when any of: Channel_id is invalid AccessError User is not part of channel	Given the channel_id of a channel that the authorised user is a part of sends authorised user's raw audio and video (if the call is a video call) data
channel/calling/getstream	GET	(token, channel_id)	{calling_buffer}	InputError when any of: Channel_id is invalid AccessError User is not part of channel User is not in call	Given the channel_id of a channel that the authorised user is a part of gets raw audio and video (if the call is a video call) data of other user(s) from the active call
channel/calling/getstatus	GET	(token, channel_id)	{time_started, call_type}	InputError when any of: Channel_id is invalid AccessError User is not part of channel User is not in call	Given the channel_id of a channel that the authorised user is a part of returns when a call was started and what type of call it is. Time_started is None if there is no active call
channel/setprofileimage	PUT	(token, channel_id, img_url)	{}	InputError when any of: img_url returns an HTTP status	Given a URL of an image on the internet, sets it as the channel image.

				<p>other than 200.</p> <p>channel_id is not valid</p> <p>Image uploaded is not a JPG</p> <p>AccessError when:</p> <p>Authorised user is not a member of channel with channel_id</p>	
user/message/send	POST	(token, u_id, message)	{message_id}	<p>InputError when any of:</p> <p>Message is more than 1000 characters</p> <p>User ID is not a valid user</p>	Send a message from an authorised user to another authorised user specified by u_id
user/messages	GET	(token, u_id, start)	{messages}	<p>InputError when any of:</p> <p>User ID is not a valid user</p> <p>Start is greater than the total number of messages in the channel</p>	Given a valid u_id returns up to 50 messages exchanged between the calling user and the user represented by u_id, with messages received being all messages in the index range start + 50

*Either an InputError or AccessError is thrown when something goes wrong. All of these cases are listed in the **Interface** table. One exception is that, even though it's not listed in the table, an AccessError is thrown when the token passed in is not a valid token.

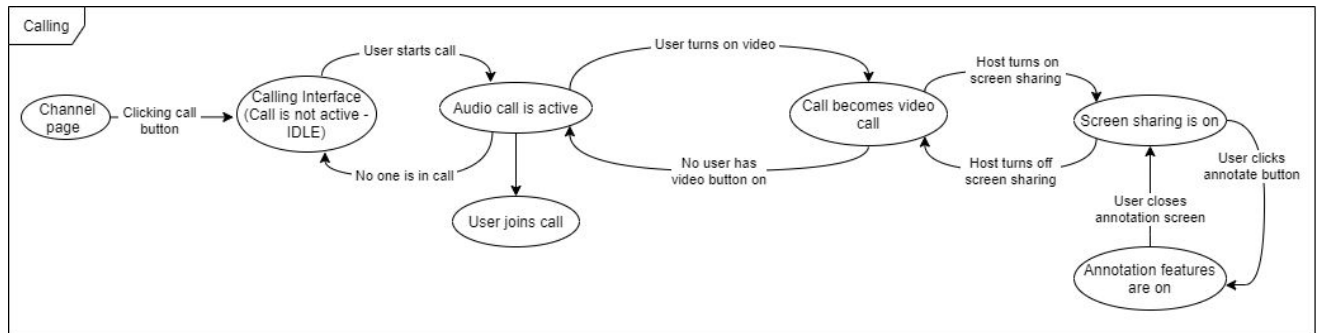
[Design] Conceptual Modelling State

Now, having a sense of the problem to solve, and well aware of the necessary capabilities required to achieve the potential solution, a state diagram was but obviously the next step to show how the state of the application would change based on user actions.

The state diagram aims to show how a developer understands the different states of the user or application.

Out of three use cases developed, state diagrams were conceptualised for two of them, that is:

Calling



Direct Messaging

