

# Executive Summary

Throughout this term, our team, YourTeam Pty Ltd, was assigned to develop an in-house digital communication tool for UNSW named Flockr. This tool was intended to support groups and teams at UNSW by providing them with a collaborative platform within the university's high intensity learning environment.

Although the core functionality of Flockr was developed based upon specifications provided to us by UNSW, it is important to consider the fact that software development is an iterative process. This involves a cycle of learning and empathising with users, followed by developing and testing hypotheses, before ultimately building features.

Consequently, our team has undertaken a process of understanding users and designing solutions to the problems they face in their collaboration and communication tools. This report will discuss how we determined the requirements for our solution based on user pain points, and provide a model for our proposed changes to the Flockr application.

## [Requirements] Elicitation:

To eliminate bias within our requirements analysis, we interviewed target users for the Flockr App, and asked them a series of comprehensive questions regarding their experience with their communication platforms. The interviews were as follows:

### Target User 1

Name: Lucia

Email: lscanlanbloor@gmail.com

*Question 1: Could you tell us what your occupation is, and in what contexts do you digitally communicate with others?*

I'm currently a first year student at university. I mainly communicate with others online because of either social or academic reasons, for example when I'm doing group assignments with others at university or having online classes, especially right now when there still aren't that many people at university.

*Question 2: What tools do you regularly use for these situations, and why do you use them?*

I mainly use Facebook Messenger or Zoom to contact others. Most of my interactions with others are video calls so these are pretty easy to use, and everyone uses them.

*Question 3: What features do you like most about these tools and what do you find annoying or frustrating?*

I really like the convenience of Facebook Messenger. It's a platform which everyone uses so there's never any difficulties in making a group chat. I think one thing which is difficult is that all my other social interactions are on there, so things can get cluttered and I can accidentally miss messages. On top of that, Facebook on the computer can be really slow which is annoying.

For Zoom, I really like the calling quality and you can share screens and break discussion up into breakout rooms, which can make talking about ideas really easy. I also like how there's an annotation feature, but it can be quite hard to use and unergonomic. I also don't like how I can't readily share and access files because you lose everything after a call, and it's also annoying how I have to keep starting up a call and inviting people every time I want to talk to others, and that each call is only 40 minutes.

*Question 4: Are there any features on these applications that you wish you could have?*

I wish Zoom could store files and that it would be easier to save groups of people you regularly talk to. I also wish going over 40 minute calls was free. I'm quite satisfied with Facebook so there's not much there.

## **Target User 2**

Name: Nirvan

Email: nirvan.chand@gmail.com

*Question 1: Could you tell us what your occupation is, and in what contexts do you digitally communicate with others?*

I'm a second year student at university currently working part-time in corporate as well. I normally communicate with others if I'm socialising with other people, communicating while doing group assignments, or having meetings and doing work for my extracurriculars at university.

*Question 2: What tools do you regularly use for these situations, and why do you use them?*

The biggest reason for me communicating online is my extracurriculars or work. For that, I'm normally using Zoom just because it's able to support large scale meetings of 10 - 30 people, which is roughly how many people there are in my weekly meetings. Other than that, for my university work I've mainly been using microsoft teams. It's the main platform for communication in a whole bunch of my university courses, and I think it's got a good balance of everything.

*Question 3: What features do you like most about these tools and what do you find annoying or frustrating?*

For Zoom, I really like the hand-up feature. Especially for large scale meetings, it's really good in providing structure to discussions. I also like the virtual backgrounds feature because it's a fun and creative way of hiding your background when you might need to in a professional environment.

For Microsoft teams, what I like the most is how you can do everything. All the file sharing is centralised, easily accessible and all microsoft applications are supported by it, and it's easy to segment different components of the work. For example in my classes, the files are sorted into things like lecture material, weekly homework, and there's channels dedicated for FAQ as well, making it really easy for me to navigate. The only problems I would have with it are the buggy/laggy calls, and that you have to download the application in order to get all the benefits (i.e. calls where you can see everyone's face, having everything load faster). I didn't want to have to download a whole application just for university work, so that's a bit annoying.

*Question 4: Are there any features on these applications that you wish you could have?*

Like I was saying with Microsoft Teams, it's a bit annoying that I have to download an application. I'm much more of a keep everything on a browser type of person. For example, I use Google Drive alot, and I enjoy that way of keeping things online and easily accessible from any device connected to the internet.

### **Target User 3**

Name: Yujia

Email: jayjaywang37@gmail.com

*Question 1: Could you tell us what your occupation is, and in what contexts do you digitally communicate with others?*

I'm a second year student at university. I normally communicate with others if I'm socialising with other people, doing group assignments, or doing society work (Currently the director in a society).

*Question 2: What tools do you regularly use for these situations, and why do you use them?*

Like most, I use Facebook and Instagram to socialise, but most regularly, I use discord. I play games alot and Discord works really well for communicating in that sense, but it actually translates really well in a group work context too. It's high quality, really really easy and intuitive to use.

*Question 3: What features do you like most about these tools and what do you find annoying or frustrating?*

I really like discord because joining and leaving calls/channels is literally a click away. There's no invite link that you have to keep going back to like Zoom, so I find that in comparison Discord is a lot more convenient. Before I got the desktop application I also really liked how I could log on as a guest and still interact with people on a web browser. The fact that Discord was so casual and easy was a big factor for why I use it so much now.

In terms of what I find difficult about it, I think the biggest thing would be that it seems as though it wasn't really intended for professional environments. The user experience, although friendly, can sometimes be a bit weird for people, and so in my group assignments others might be dissuaded from using it. I think another thing is that I have to scroll back and forth just to find files, photos or videos in a channel because it's not stored in a convenient tab like Facebook is. With that, it's a bit annoying how voice channels and message channels are distinct. I organise things by association, so I'd like to see when I had calls and what messages and files are associated with those calls, rather than having to open up Word or Google Docs and take minutes.

*Question 4: Are there any features on these applications that you wish you could have?*

Discord is overall pretty great, but as I said, there's no easy file repository which would be really great. On top of that, I wish I could see when people were last active like it is on Facebook. It's not a huge pain point, but I think it'd be good for transparency in teamwork.

## [Requirements] Analysis & Specification - Use Cases

We then consolidated our learning into user stories as follows, focusing the stories and cases which caused interviewees the most frustration.

### User Story 1: Lucia

User Acceptance Criteria

**As a** student,

**I want to** be able to store my files after the call ends,

**so that** I can keep organised, be able to access it whenever and be ensured that the files are safe.

**Use Case:** Store a file

**Goal in Context:** Users need to be able to store a file within the application need to be able to create an account by providing details about their first and last name, their email address and password meeting requirements.

**Scope:** Flockr Application

**Level:** Primary Task

**Preconditions:** The Flockr application is up and running regularly without errors, the user has also created an account and is logged in

**Success End Condition:** The user finds a local file and sends it to the channel they are in, which should then be accessible to anyone in the channel

The user would have created their account and will be redirected inside the website

**Fail End Condition:** The user will be unable to send a file into the channel

The user will not create an account and be stuck on the signup page

**Primary Actor:** User tester

**Trigger:** User clicks the 'attach a file' button

### **Main Success Scenario**

Step 1. User is already registered and logged into Flockr

Step 2. User clicks the 'attach a file' button

Step 3. User selects a file they want to attach

Step 4. The user successfully sends the file to the channel

Step 5. The file remains there forever until it is deleted by either the user who sent it, the owner of the channel, or the owner of the flockr

User Story 2:

User Acceptance Criteria

**As a** part-time corporate worker ,

**I want to** be able to access my video calls via web browsers rather than application

**so that** it will be much more easy to use especially without needing to download a lot of different applications, thus, saving me time in a very busy schedule, and freeing up my computer space.

Use Case (User 2):

**Use Case:** Access a call via browser

**Goal in Context:** Users need to be able to access a call via the internet browser instead of the application

**Scope:** Flockr Website

**Level:** Primary Task

**Preconditions:** The Flockr website is up and running regularly without errors, the user has also created an account and is logged in

**Success End Condition:** The user finds their call via the Flockr website, and joins successfully

**Fail End Condition:** The user cannot access the webpage and thus, cannot enter the call

**Primary Actor:** Common user

**Trigger:** User accesses the Flockr website

### **Main Success Scenario**

Step 1. User accesses the Flockr website via typing in the url or clicking the website from a search

Step 2. User joins the channel which the call is currently occurring in

Step 3. User has all the same functionalities as they would have calling on Flockr via the application

### User Story 3:

User Acceptance Criteria

**As a** society director

**I want to** be able to access all the files if need be in a specific repository

**So that** it will be very efficient for when trying to access files which were sent a while back, not needing to scroll through messages to find it

*No centralised file repository*

*Message channel and voice channel are separated*

*Last active feature*

Use Case (User 3):

**Use Case:** Find a file

**Goal in Context:** Users need to be able to find a file within a repository based system

**Scope:** Flockr Application

**Level:** Primary Task

**Preconditions:** Files can be sent successfully, and the repository is working with all the files being saved in it

**Success End Condition:** The user clicks on the file repository button and can see all the files which were sent in the channel before, and can access the file too

**Fail End Condition:** The user will be unable to click on the repository button, and thus, cannot see any of the previous files which were sent in the channel

**Primary Actor:** Common user

**Trigger:** User clicks the “file repository” button

### **Main Success Scenario**

Step 1. User clicks on the “file repository” button

Step 2. The user can see all the files which were previously sent in the channel

Step 3. The files have all the required information along with it, such as who sent it and the time it was sent

Step 4. User finds their wanted file, and can download it by clicking on it

### **[Requirements] Validation**

To ensure that our consolidation of the interviewees experience with communication platforms was an accurate summation of their experiences, we briefly interviewed them again, asking the following:

1. *“Do you think that this user story accurately reflects you, and in particular, your pain points with the tools you currently use?”*
2. *“Would you be more satisfied with Flockr over some of your current communication tools, if the Main Success Scenario we’ve provided you was achieved?”*

### **Target User 1 - Lucia**

1. Yes definitely, it’s always a bit embarrassing when I have to go back and ask my group chats for files after the Zoom call has ended
2. Yes, I think what’s quite difficult right now with file sharing is I have to use something like google drive alongside communication tools. Or I have to use Facebook which gets me distracted. This seems like an easy enough process.

### **Target User 2 - Nirvan**

1. For sure. Unfortunately my workplace doesn’t provide me with a work laptop, so I have to do everything on just one laptop. I normally organise everything in folders on chrome, because if I have too many applications my laptop slows down.
2. Maybe, Flockr seems quite convenient, and it seems like I’m getting everything it has to offer, without me needing to go the extra mile. But I’d say that Microsoft Teams might still be a better alternative for me, just because of the extent of work I need to do with Microsoft Office Applications.

**Target User 3 - Yujia**

1. Yes! I think this collaborative component with the files embedded in the teamwork process is something which I've been missing in my society work. There are ways around it right now, but I wish these features existed.
2. It would definitely be very tempting. It would be a great way to integrate and sort my files into their respective teams without me needing to manually organise it.



## [Design] Interface Design

We then created a table of routes which would be our proposed future implementations.

Route	Method	Params	Return	Description
channel/attachfile	POST	{token, channel_id, file, time_sent}	{file_id}	User sends a file into the channel. It also records who sent it, and what time it was sent. Assumption that files can only be word docs and pdf
channel/filerepository	POST	{token, channel_id, file_repository}	{files}	User clicks the 'file repository' button and accesses all the files which were sent in that channel. It will also be able to see the time when the file was sent, and who sent it. The files will be listed out in a table format and will be compact to save space and be easy to access.
filerepository/page	POST	{file_repository_id, page_number}	{page_number}	Users can decide to view a new page depending on whether or not the file repository has enough files for it. Assumption that there will be a maximum of 10 pages for the file repository
filerepository/filedetails	GET	{file_repository_id, file}	{u_id, time_sent}	Users can check the details of the file, which will show the u_id of the user who sent the file, and the time that it was sent. User can then also download the file from this

webpage/channel/call	POST	{is_browser, token, channel_id}	{}	Users can join a call via the web browser of Flockr, instead of the application. It would function the same way as joining a channel, then joining a call, which is how it would work on the application
----------------------	------	------------------------------------	----	--

## [Design] Conceptual Modelling (State)





