

## **REQUIREMENTS]: ELICITATION**

### **QUESTIONS:**

1. Do you use teamwork driven communication tools (such as Discord, Slack, Microsoft Teams, etc)?
  - a. If so, why do you use it over the other applications? Are there certain features you prefer on your chosen application?
2. What feature do you least like about your application? Is there a specific reason as to why?
  - a. What ways do you think this feature could be improved?
3. What problems have you run into with your application?
  - a. Has there been a feature that you've found yourself needing that the application has been lacking in? If so, what is it?
4. What feature would you like to see implemented in your application? Any particular reason why?
5. What feature do you think is the most important part about your application? Why?

### **PERSON 1: Melinda**

**Email:** mliu236@gmail.com

1. Yes, I use slack, discord, zoom - most of these I started using as a result of friends or uni - I prefer these over other applications as they're more commonly used thus easier to connect to others and have satisfactory features (e.g. screen share, tagging, bots)
2. Nothing particularly other than it requires strong internet of u have a lot of things running at once e.g screen share and video call
  - a. um.... 6G?
3. At times, my discord creates an echo but I think that's more of my internet problem than discord's problem. On Slack, video call sucks. A lot. You aren't able to see everybody at once, and the connection is really bad, so that is a feature that can be improved and made stronger
4. A discord feature could be multiple screen share / screen view - (being able to view multiple shared screens at once) that would be cool and useful to compare between team members / when you're just playing games
5. The video call and being able to see multiple people at once (gallery view on zoom) - esp in virtual meetings it makes the atmosphere feel a lot more whole and not as isolating

### **PERSON 2: Mariah**

**Email:** mariah.eleni2001@gmail.com

1. Yes, I primarily use Zoom. I only use it over other applications because I am obligated to through school.
2. My least favourite feature is the time limit on group calls with three or more people. After 40 minutes, the call ends and members are kicked out of the call. The unlimited version is behind a paywall, which is not an issue with services like Skype or Discord who offer group calls for free. Given that Zoom's competition doesn't have such a paywall, I

believe it's an unnecessary hurdle, especially during these times where online learning was so crucial.

3. Every Zoom call I have personally been in has been met with technical issues. I assume everyone in the call would need outstanding internet connection, as every call I have been in has involved teachers randomly dropping out/ cutting off.
4. Unlike Skype and Discord, Zoom does not have an option to create a forum for permanent text messages. I'm aware that Zoom is simply just used for calls/meetings, but when there are applications such as Discord that offer so much more than Zoom, it feels very lacklustre. As Zoom became the default application for online learning, it was quite frustrating knowing there were better applications out there while we were using Zoom.
5. Group meetings are the most important feature of Zoom as it is its sole purpose.

### **PERSON 3: Jason**

**Email:** jasonjxtan@gmail.com

Main collaborative software: Discord

Reason for use above other softwares:

- Most frequent/abundant software within peer group - not everyone has slack or MS Teams but most have discord which is easier to use/implement for meetings
- Easiest to setup as an attendee - as long as you are in the server it's a simple click to join and leave
- Overlay is the easiest to use, and unlike other softwares can be used in the background of other tasks easier

Difficulties:

- Sometimes hard to "find" certain things in channels (search tool is a little slow + harder to understand relative to e.g messenger. Even harder to find files or images, so unless you know an exact time/phrase its pretty hard

Feature/s for implementation:

- would be greatly beneficial to implement a more collaborative UI for workplace communications - e.g file sharing/storage interface or folders or something
- Ability to make certain bots channel specific to avoid 'spam' during someone else's usage

Most important Discord feature;

- ability to call and chat with others in an environment where it's easy to switch between calls/ easy access in and out of a call/chat/server etc.

## **[REQUIREMENTS]: ANALYSIS AND SPECIFICATIONS - USE CASES**

**PERSON 1: Melinda**

### **User Story:**

As a student who often participates in group discussions via video calls, I want the ability to see everybody on my screen at once instead of having the screen focus on the main speaker, so that I can better communicate with my team members.

### **Rule-Based User Acceptance Criteria:**

- Call function exists within the application
- A call can be started by clicking the call button within a channel
- Participants of the channel the call has been started in can join the call
- The option to turn face cam on can be available for each user
- A button exists that can change the view of participants in the call (from main speaker only to gallery view)
- The maximum amount of participants that can be viewed at once with gallery view is 25 per page

### **Scenario Oriented Acceptance Criteria:**

**Scenario:** A video call has been started

**Given:** There are 4 participants within the video call with their cameras on

**When:** The user has selected the option to change video view

**And:** The current view is set on main speaker focus

**Then:** The view will toggle to a gallery view, where all 4 participants can be seen at once

### **Use Case:**

**Use Case:** See all participants in a video call simultaneously

**Goal In Context:** When there are more than two people in a video call with their cameras on, there exists an option to toggle the view from main speaker focus to gallery view

**Scope:** Channel, video call, participants of the video call

**Level:** Primary task

**Preconditions:** There are at least two registered users in a video call with their cameras on, and the view is set on main speaker focus

**Success End Condition:** The option to toggle to gallery view has been triggered, and now the videos of all the participants in the video call can be seen on the screen at once

**Failed End Condition:** The option to toggle to gallery view has not been or cannot be triggered, and the view remains on main speaker focus

**Primary Actor:** Registered user who is participating in a video call

**Trigger:** A video call is initiated by pressing the call button, and at least three users enter the call with cameras on

### **Main Success Scenario:**

1. A video call is initiated by pressing the call button

2. At least three registered users join the video call
3. The current view is set to main speaker focus
4. The option to toggle to gallery view is triggered, and the user who triggered the option can now see all speakers in gallery view

## **PERSON 2: Mariah**

### **User Story:**

As Mariah, a user, I would like to communicate seamlessly for long periods of time for free with my peers and teachers, so I can better understand class-work

### **Rule-Based User Acceptance Criteria:**

- Group calls are available from the 'channel' page
- Group calls can be held with existing channels
- As many people as needed can be added to group calls
- Group calls will run infinitely

**Scenario:** Would like to initiate a group calls with 3 others

**Given:** All users are registered

**When:** The user selects the 'call' button on a channel

**And:** At least one other member accepts

**Then:** A group call will be initiated

**And:** The call will run infinitely with no time limit

**Use Case:** Initiate group call

**Goal in context:** Users need to be able to hold group calls with other users with no time limit

**Scope:** Channel, users

**Level:** Primary task

**Preconditions:** All users are registered

**Success End Condition:** A group call is initiated with all members of the channel that will run for as long as desired by the members

**Failure End Condition:**

- A group call is not initiated
- A group call is initiated but ends prematurely

**Primary Actor:** User - The individual that initiates the group call

**Trigger:** User initiates a group call by clicking the 'call' button on the channel

## **MAIN SUCCESS SCENARIO**

1. User clicks onto the desired channel and clicks the 'call' button
2. A call is initiated and all members of the channel are notified
3. Members receive the notification and accept the call
4. Users enter the call
5. The call is active until all members leave

### **PERSON 3: Jason**

#### **User Story:**

As a user, I would like to be able to find files that were previously sent more easily so I can better save my time.

#### **Rule-Based User Acceptance Criteria:**

- A registered user who is a participant of the channel sends a file
- The file is added to a storage folder within the channel
- User (who is part of the channel) can access the storage folder via a button
- They can search for the file needed using search terms or details about the file (date sent, file type, etc.)

#### **Scenario Oriented Acceptance Criteria:**

**Scenario:** User is searching for a file previously sent into the channel

**Given:** User is registered and a participant of the channel, the file has been sent at some point during the history of the channel

**When:** User opens the storage folder via a button on the interface

**And:** Searches for the file needed via a key term (its name)

**Then:** The system will return all results (files) that match the key term

#### **Use Case:**

**Use Case:** Search for a file

**Goal In Context:** User searches for a file that's been sent into the channel by using the search function within the storage folder

**Scope:** Channel, registered users, files that've been sent

**Level:** Primary task

**Preconditions:** The user is a registered user who is a participant of this channel, the file they're searching for exists and has been sent into this channel at some point

**Success End Condition:** When the key term is entered into the search bar, the system returns all files that match the key term, with at least one of these files being the file the user was searching for.

**Failed End Condition:** The user is not registered, the file does not exist in the storage folder, the key term fails to return results

**Primary Actor:** Registered user who is searching for a file

**Trigger:** When the storage folder button is clicked

#### **Main Success Scenario:**

1. Registered user of the channel clicks on the storage folder button located within the channel
2. The storage folder is opened up with a search bar
3. User can either:
  - a. scroll through the storage folder and view all previous files sent into channel or

- b. search for the needed file via search terms (name, date range that the file was sent, file type..). This option will search through all files that've been sent in the server in channels the user has access to
4. System returns all results that match the search terms, with at least one of these files being the one the user needed

## **[REQUIREMENTS]: VALIDATION**

**Melinda:** 'I think this additional feature would be highly beneficial in improving user experience with the Slack app and reducing the issues raised when speaker view was the only video call option.'

**Mariah:** 'The problem has been described perfectly'

**Jason:** 'This is able to solve my existing problem by providing an improved interface regarding file storage and retention'

## **[DESIGN]: INTERFACE DESIGN**

### **PERSON 1: Melinda**

<b>Name &amp; Description</b>	<b>HTTP Method</b>	<b>Data types</b>	<b>Exceptions</b>
channel/call/v1  Initiate a call in the channel with the given channel_id	POST	<b>Parameters:</b> (token, channel_id) <b>Return Type:</b> {}	<b>InputError</b> when any of: <ul style="list-style-type: none"><li>- Channel ID is not a valid channel</li></ul> <b>AccessError</b> when <ul style="list-style-type: none"><li>- Authorised user is not a member of channel with channel_id</li></ul>
channel/joincall  Given a channel id, join the active call in this channel	POST	<b>Parameters:</b> (token, channel_id) <b>Return Type:</b> {}	<b>InputError</b> when any of: <ul style="list-style-type: none"><li>- Channel ID is not a valid channel</li></ul> <b>AccessError</b> when <ul style="list-style-type: none"><li>- Authorised user is not a member of channel with</li></ul>

			channel_id
channel/call/view  Given a channel id, toggles the view for the authorised user between main speaker focus and gallery view.	PUT	<b>Parameters:</b> (token, channel_id)  <b>Return Type:</b> {}	<b>InputError</b> when any of: <ul style="list-style-type: none"> <li>- Channel ID is not a valid channel</li> </ul> <b>AccessError</b> when <ul style="list-style-type: none"> <li>- Authorised user is not a member of channel with channel_id</li> <li>- Authorised user is not a participant of a call</li> </ul>

## PERSON 2: Mariah

Name & Description	HTTP Method	Data types	Exceptions
channel/call/v1  Initiate a call in the channel with the given channel_id	POST	<b>Parameters:</b> (token, channel_id) <b>Return Type:</b> {}	<b>InputError</b> when any of: <ul style="list-style-type: none"> <li>- Channel ID is not a valid channel</li> </ul> <b>AccessError</b> when <ul style="list-style-type: none"> <li>- Authorised user is not a member of channel with channel_id</li> </ul>
channel/leavecall  Given a channel id, the user is removed from the call active in the channel. Note: If user is the last member in the call, the call is ended	POST	<b>Parameters:</b> (token, channel_id)  <b>Return Type:</b> {}	<b>InputError</b> when any of: <ul style="list-style-type: none"> <li>- Channel ID is not a valid channel</li> </ul> <b>AccessError</b> when <ul style="list-style-type: none"> <li>- Authorised user is not a member of channel with channel_id</li> </ul>

			<ul style="list-style-type: none"> <li>- Authorised user is not currently in the active call</li> </ul>
<p>channel/endcall</p> <p>If the user initiated the call, remove all users from the active call and end it.</p>	POST	<p><b>Parameters:</b> (token, channel_id)</p> <p><b>Return Type:</b> { }</p>	<p><b>InputError</b> when any of:</p> <ul style="list-style-type: none"> <li>- Channel ID is not a valid channel</li> </ul> <p><b>AccessError</b> when</p> <ul style="list-style-type: none"> <li>- Authorised user is not a member of channel with channel_id</li> <li>- Authorised user did not initiate the call</li> </ul>
<p>channel/joincall</p> <p>Given a channel id, join the active call in this channel</p>	POST	<p><b>Parameters:</b> (token, channel_id)</p> <p><b>Return Type:</b> { }</p>	<p><b>InputError</b> when any of:</p> <ul style="list-style-type: none"> <li>- Channel ID is not a valid channel</li> </ul> <p><b>AccessError</b> when</p> <ul style="list-style-type: none"> <li>- Authorised user is not a member of channel with channel_id</li> </ul>

### PERSON 3: Jason

Name & Description	HTTP Method	Data types	Exceptions
<p>message/sendfile</p> <p>Send a file from the authorised user to the channel specified by channel_id, and adds the file to the storage folder.</p>	POST	<p><b>Parameters:</b> (token, channel_id, file)</p> <p><b>Return Type:</b> {message_id}</p>	<p><b>InputError</b> when any of:</p> <ul style="list-style-type: none"> <li>• File size is larger than 8mb</li> <li>• Channel ID is not a valid channel</li> </ul>

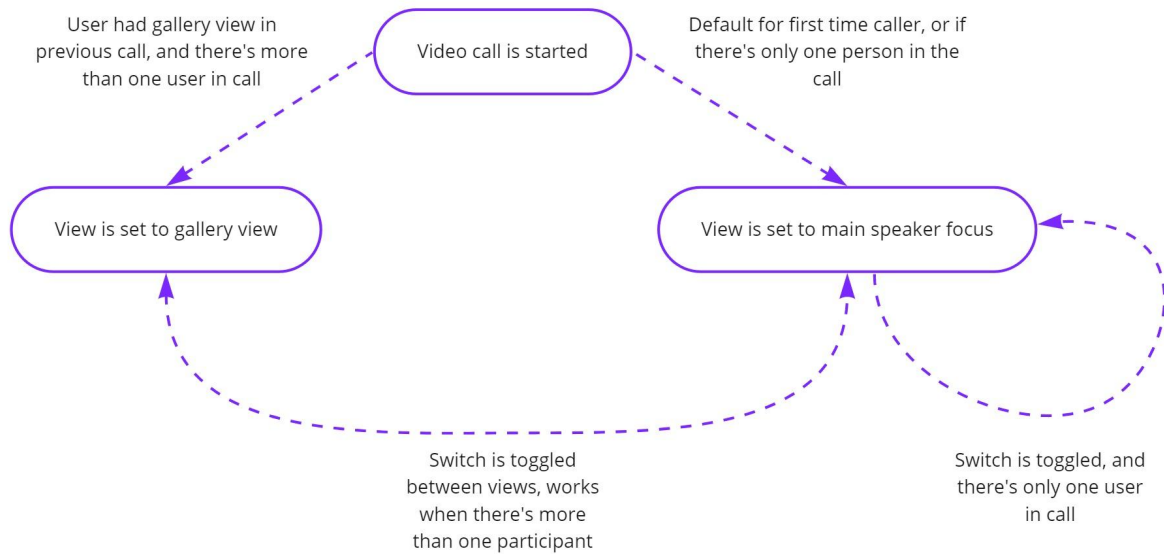


			<b>AccessError</b> when: <ul style="list-style-type: none"> <li>the authorised user has not joined the channel they are trying to post to</li> </ul>
<p>message/removefile</p> <p>Given a message_id for a message, this message is removed from the channel/DM, and the corresponding file is removed from the storage folder.</p>	DELETE	<p><b>Parameters:</b> (token, channel_id, message_id)</p> <p><b>Return Type:</b> { }</p>	<p><b>InputError</b> when any of:</p> <ul style="list-style-type: none"> <li>Message (based on ID) no longer exists</li> </ul> <p><b>AccessError</b> when none of the following are true:</p> <ul style="list-style-type: none"> <li>Message with message_id was sent by the authorised user making this request</li> <li>The authorised user is an owner of this channel (if it was sent to a channel) or the <b>**Dreams**</b></li> </ul>
<p>channel/files</p> <p>Given a Channel with ID channel_id that the authorised user is part of, return up to 20 files between index "start" and "start + 20". File with index 0 is the most recent file in the channel. This function returns a</p>	GET	<p><b>Parameters:</b> (token, channel_id, start)</p> <p><b>Return Type:</b> { files, start, end }</p>	<p><b>InputError</b> when any of:</p> <ul style="list-style-type: none"> <li>Channel ID is not a valid channel</li> <li>start is greater than the total number of files in the channel</li> </ul> <p><b>AccessError</b> when</p>

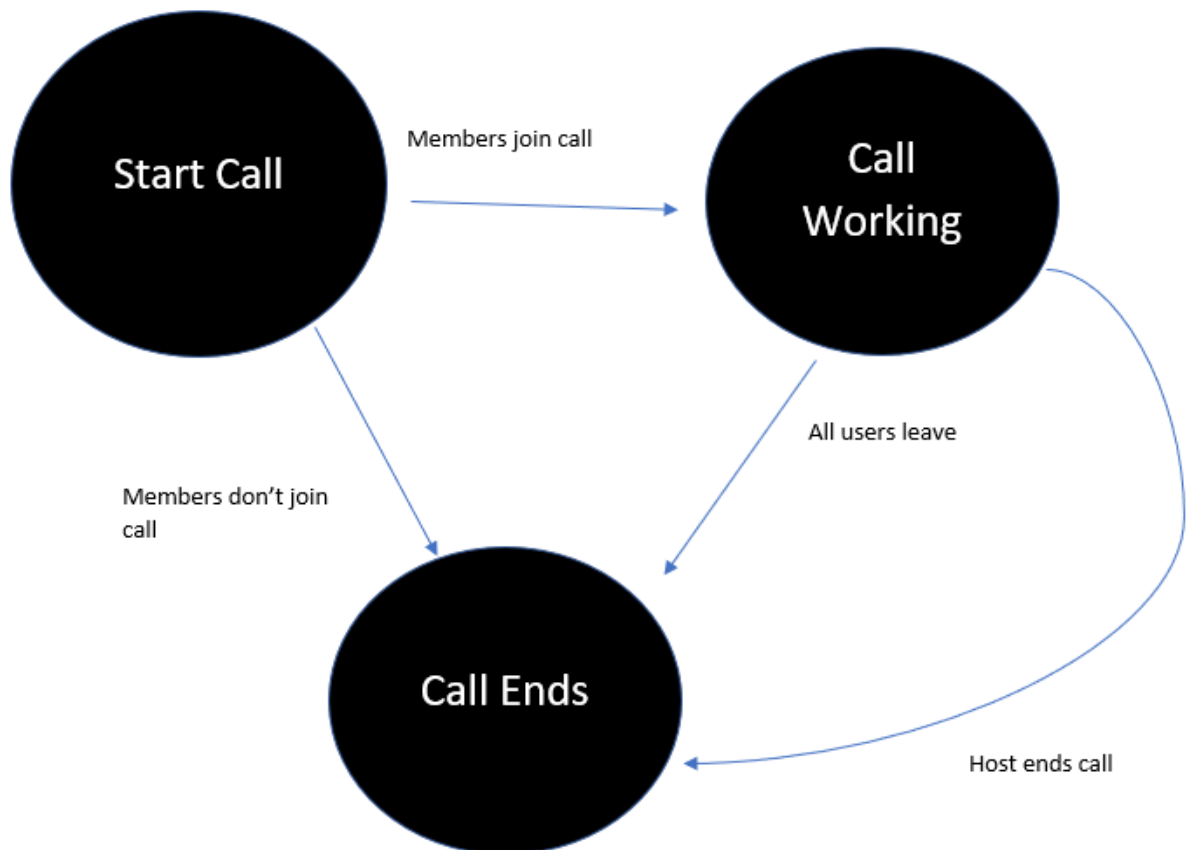
<p>new index "end" which is the value of "start + 20", or, if this function has returned the least recent files in the channel, returns -1 in "end" to indicate there are no more files to load after this return.</p>			<p><b>any of:</b></p> <ul style="list-style-type: none"> <li>• Authorised user is not a member of channel with channel_id</li> </ul>
<p>channel/storage</p> <p>Provides all files that have been sent into the channel with ID channel_id</p>	GET	<p><b>Parameters:</b> (token, channel_id)</p> <p><b>Return Type:</b> { files }</p>	<p><b>InputError when any of:</b></p> <ul style="list-style-type: none"> <li>• Channel ID is not a valid channel</li> </ul> <p><b>AccessError when any of:</b></p> <ul style="list-style-type: none"> <li>• Authorised user is not a member of channel with channel_id</li> </ul>
<p>search/files</p> <p>Given a query string, return a collection of files in all of the channels/DMs that the user has joined that match the query</p>	GET	<p><b>Parameters:</b> (token, query_str)</p> <p><b>Return Type:</b> { files }</p>	<p><b>InputError when any of:</b></p> <ul style="list-style-type: none"> <li>• query_str is above 1000 characters</li> </ul>

## [DESIGN]: CONCEPTUAL MODELLING

### PERSON 1: Melinda



### PERSON 2: Mariah



### PERSON 3: Jason

