# Planning

## Requirements and Design

By Farhan Chowdhury for

**H17A_ELEPHANT**

# Elicitation

**Interview I: Croy Simson (c.simpson124@gmail.com)**

***Q1: What teamwork-driven communication tool or app do you use the most?***

A: I use discord a lot.

***Q2: What are your favourite things about discord?***

A: I find discord very convenient to use. Discord allows me to message and call my friends and team members very quickly. I am able to see when people are online and what they are doing. Simply, it allows me to reach out to team members and keep updated with the projects I work on without much hassle.

***Q3: What are some problems you encounter with discord?***

A: Since I work on software projects, it becomes pretty difficult to transfer files of code through discord due to file size limits that are placed. In order to increase the limit, you would need to purchase a monthly subscription which I don't find appealing in the slightest. It would improve the workflow by a lot if we were able to transfer code files, when helping each other debug for example, as screenshots make it hard to try the code ourselves.

**Interview II: Caitlyn Eliza (elizac584@outlook.com)**

***Q1: What teamwork-driven communication tool or app do you use the most?***

A: Microsoft Teams

***Q2: What are some of your favourite things about teams?***

A: I like how organised Microsoft Teams is for my workflow. I have separate channels for all my different teams and group projects. It is very easy to call and message people. There is also a calendar that makes it easy to organise meetings and keep me on top of my assignments and due dates so I don't forget anything.

***Q3: What are some problems that you have encountered with teams?***

A: Since I do a lot of programming and mathematics, I dislike teams as it does not let me format code and indentations properly. To get around it I either have to upload my files or send screenshots which either makes things very cluttered or hard to try my code. Also, when it comes to mathematics, I cannot type any equations or mathematical notation … this makes it very hard for me to ask questions or help others with solutions.

**Interview III: Josh Engert (joshmengert@gmail.com)**

***Q1: What teamwork-driven communication tool or app do you use the most?***

A: I use discord the most

***Q2: What are your favourite things about discord?***

A: Discord makes it very easy for me to message and call my peers. It's as simple as one button to call someone. For my group based project, we usually create channels with voice chats where we can hop on whenever we are free. When someone needs help, me or someone else in the group would usually go on a voice chat and help solve the issue while the other person shares their screen.

***Q3: What are some problems you encounter with discord?***

A: Usually in a team environment, there are a lot of decisions that need to be made with everyone's input. There isn't really anything on discord that helps with this issue. It'd be great if we were able to make a poll as easily as sending emojis or gifs. I think it'll make the workflow smoother.

# Analysis and Specification

## Interview I

**User story 1:**

As a software developer,
I would like to be able to send and transfer files without a strict limit, so that others can use my files or help debug them on their own machine conveniently.

**User Acceptance Criteria 1**

- The “Attach File” button is placed near the message text box.
- Users are able to attach a file once they click the “Attach File” button.
- When “Attach File” is clicked, there is a pop-up window where the users can drag and drop the file they wish to send.
- The pop-up contains the two buttons “send” and “cancel”, allowing users to send the drag-and-dropped files or cancel their action respectively.

**Use Case 1**

1. User clicks “Attach File”
2. A window pops up asking the user to “Drag and drop files here”, as well as containing buttons to “Send” or “Cancel”
3. User drags and drops the files they wish to send
4. Files are loaded in the window
5. User clicks “Send”
6. All files dragged and dropped in the pop-up window are sent in the channel

## Interview II

**User story 2:**

As a mathematics student,
I would like to be able to type and send equations and mathematical notations, so I am able to send others questions or solutions: when I need help, or when there is someone else I can help respectively.

**User Acceptance Criteria 2**

- There is a text command “!latex”
- When users type “!latex” at the beginning of their message, anything after the command is expected to be latex input.
- When the user finishes typing and presses enter or send, their message is compiled and is output as latex.

**Use Case  2**

1. User clicks the message text box
2. User types “!latex” first
3. User continues to type in the latex language as if they are creating a latex document
4. User enters the message by clicking send
5. Latex output is sent

**User story III:**

As a programmer,
I would like to type with indententations and properly formatted code in my messages, so that I can send my code to others for them to try or debug (for work purposes).

**User Acceptance Criteria 3**

- There is a text command → ``` *insert code* ```
- Anything written in between the 3 single quotes will maintain the exact indentation input by the user once sent

**Use Case 3**

1. User clicks the message text box
2. User enters ``` ```
3. User copies their code from their preferred text editor and pastes it in between the triple single quotes OR user types in between the triple single quotes
4. User clicks “send”
5. The indentation and formatting used in between the triple single quotes are maintained in the message after sending.

# Interview III

**User story 4:**

As a person working in a team environment,
I would like to create polls so that we are able to make collective decisions.

**User Acceptance Criteria 4**

- Next to the message text box, there is a cog symbol for options
- Clicking the cog opens a drop down menu, with a “create poll” option
- Clicking poll opens up a menu containing “Poll title”, and three options named “option 1”, “option 2” and “option 3” as text boxes for editing.
- There is a plus button below the options for adding additional options.
- At the bottom, there is a “submit poll” button for sending the poll to the channel.
- Once the poll is submitted, users are able to select an option.
- Once an user selects an option, that user is able to see the results of the poll.
- There is a “view results” button below the poll that shows the results of the poll as well.

**Use Case 4**

1. User clicks cog
2. A menu drops down, with one of the features being “create a poll”
3. User clicks “create a poll”
4. A menu opens up containing “Poll title”, and three options named “option 1”, “option 2” and “option 3” as text boxes for editing. There is a plus button below the options and a “submit poll” option.
5. User inputs necessary text for “Poll title”, “option 1”, “option 2” and “option 3”.
6. User clicks the plus button.
7. Another option is created, “option 4”
8. User clicks “option 4” and edits with necessary option
9. User clicks “submit poll”
10. The poll is sent to the channel.

# Validation

Each user from the target users interviewed above were shown the use case related to the problems they encountered while using their preferable teamwork-driven communication tool. Afterwards, they were asked the following questions.

**Validation from Croy Simson (c.simpson124@gmail.com)**

***Q: Does the following scenario (use case 1) solve the problems you encounter with sending files?***

A: Yes, this seems like a very simple solution to the problem I've been having.

***Q: Is there anything else you would like to add?***

A: This is incredibly picky but a way to make it a tad more efficient would be to allow users to just drag and drop files into the message box straight away. But regardless, the scenario you gave me seems fine.

**Validation from Caitlyn Eliza (elizac584@outlook.com)**

***Q: Does the following scenario (use case 2) solve the problems you encounter with typing mathematical notation?***

A: Yes, since I am a mathematics student I use latex a lot for my assignments so this is a neat addition.

***Q: Is there anything else you would like to add?***

A: Just a little concerned for people who do not have much experience using latex. They might find it difficult to use the functionality.

***Q: What do you think about this (use case 3)? Does this solve the problems you encounter as a programmer when sending code to others?***

A: I think this is a really good solution to the problem. This will definitely make it much easier to send code to each other on the fly.

***Q: Is there anything else you would like to add?***

A: Nope, nothing else. Thanks.

**Validation from Josh Engert (joshmengert@gmail.com)**

***Q: Does the following scenario (use case 4) solve the problems you encounter with making collective group decisions?***

A: Yes, polls are a great idea. This will make forming collective decisions in a team environment much easier.

***Q: Is there anything else you would like to add?***

A: No.

# Interface Design

| **Name & Description** | **HTTP Method** | **Data Types** | **Exceptions** |
|---|---|---|---|
| file/send/v1<br><br>Send a file from the authorised user to the channel specified by channel_id. Note: Each file should have its own unique ID, i.e. no file should share an ID with another file, even if that other message is in a different channel. | POST | **Parameters:**<br>{ token, channel_id, file}<br><br>**Return Type:**<br>{ file_id } | **InputError** when:<br>● Channel_id does not refer to a valid chanel<br>● Size of file is less than 0.125 bytes<br>**AccessError** when:<br>● Channel_id is valid and the authorised user is not a member of the channel |
| message/send/latex/v1<br><br>If the message in message/send/v1 contains “!latex” as the first word, send the string after “!latex” from the authorised user to the channel specified by channel_id as latex output. Note: Each message should have its own unique ID, i.e. no messages should share an ID with another message, even if that other message is in a different channel. | POST | **Parameters:**<br>{ token, channel_id, message }<br><br>**Return Type:**<br>{ message_id } | **InputError** when:<br>● Channel_id does not refer to a valid channel<br>● Length of message is less than 1 or over 1000 characters<br>**AccessError** when:<br>● Channel_id is valid and the authorised user is not a member of the channel |
| maths/send/v1<br><br>Input the mathematics symbol associated with the maths_id in the message. | POST | **Parameters:**<br>{ token, maths_id }<br><br>**Return Type:**<br>{} | **InputError** when:<br>● Maths_id is not a valid maths symbol id |
| poll/send/v1<br><br>Send a poll from the authorised user to the | POST | **Parameters:**<br>{ token, channel_id, poll_name, options, timeframe } | **InputError** when:<br>● Channel_id does not refer to a valid channel |

| | | | |
|---|---|---|---|
| channel specified by channel_id. Note: Each poll should have its own unique ID, i.e. no polls should share an ID with another poll, even if that other poll is in a different channel. | | **Return Type:**<br>{ poll_id } | ● Length of poll_name is less than 1 or over 1000 characters<br>● Timeframe is less 1<br>● Length of options is less than 1<br>**AccessError** when:<br>● Channel_id is valid and the authorised user is not a member of the channel |
| poll/close/v1<br><br>Given a poll created by the authorised user, close the poll so that users are no longer able to vote | POST | **Parameters:**<br>{ token, poll_id}<br><br>**Return Type:**<br>{} | **InputError** when:<br>● Poll_id does not refer to a valid poll<br>**AccessError** when:<br>● Poll_id is valid and the authorised user did not create the poll. |
| message/send/indent/v1<br><br>If the message in message/send/v1 contains ``` ``` wrapped around a string, send the string in between the triple single quotes from the authorised user to the channel specified by channel_id in indentation mode. Note: Each message should have its own unique ID, i.e. no messages should share an ID with another message, even if that other message is in a different channel. | POST | **Parameters:**<br>{ token, channel_id, message }<br><br>**Return Type:**<br>{ message_id } | **InputError** when:<br>● Channel_id does not refer to a valid channel<br>● Length of message is less than 1 or over 1000 characters<br>**AccessError** when:<br>● Channel_id is valid and the authorised user is not a member of the channel |

# Interface input/output types

| **Variable name** | **Type** |
|---|---|
| Has suffix **id** | integer |
| Named exactly **message** | string |
| Contains substring **name** | string |
| Named exactly **timeframe** | Integer (represented in seconds) |
| Named exactly **file** | Normal text files |
| Named exactly **token** | **s**tring |
| Named exactly **options** | List of strings |

# Conceptual Modelling

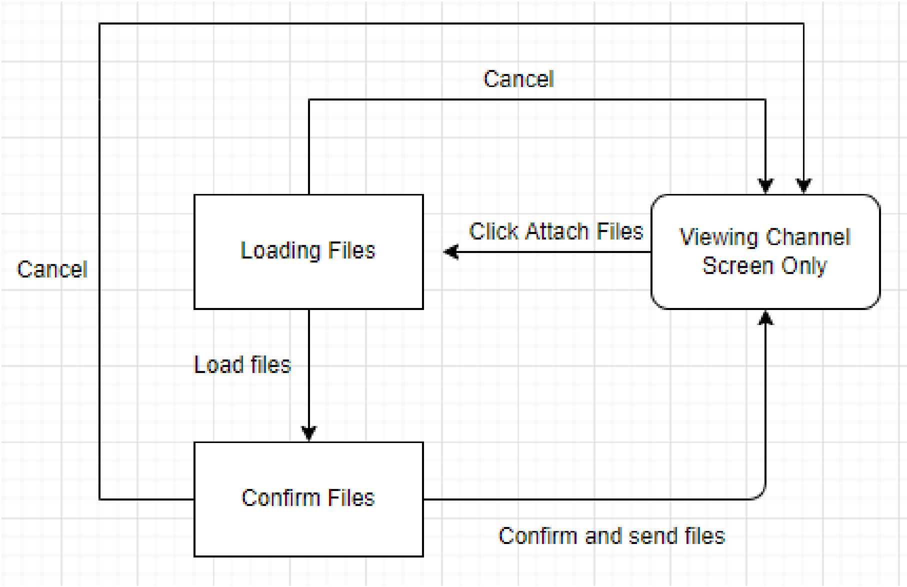

Figure 1: Conceptual model of **file/send/v1**

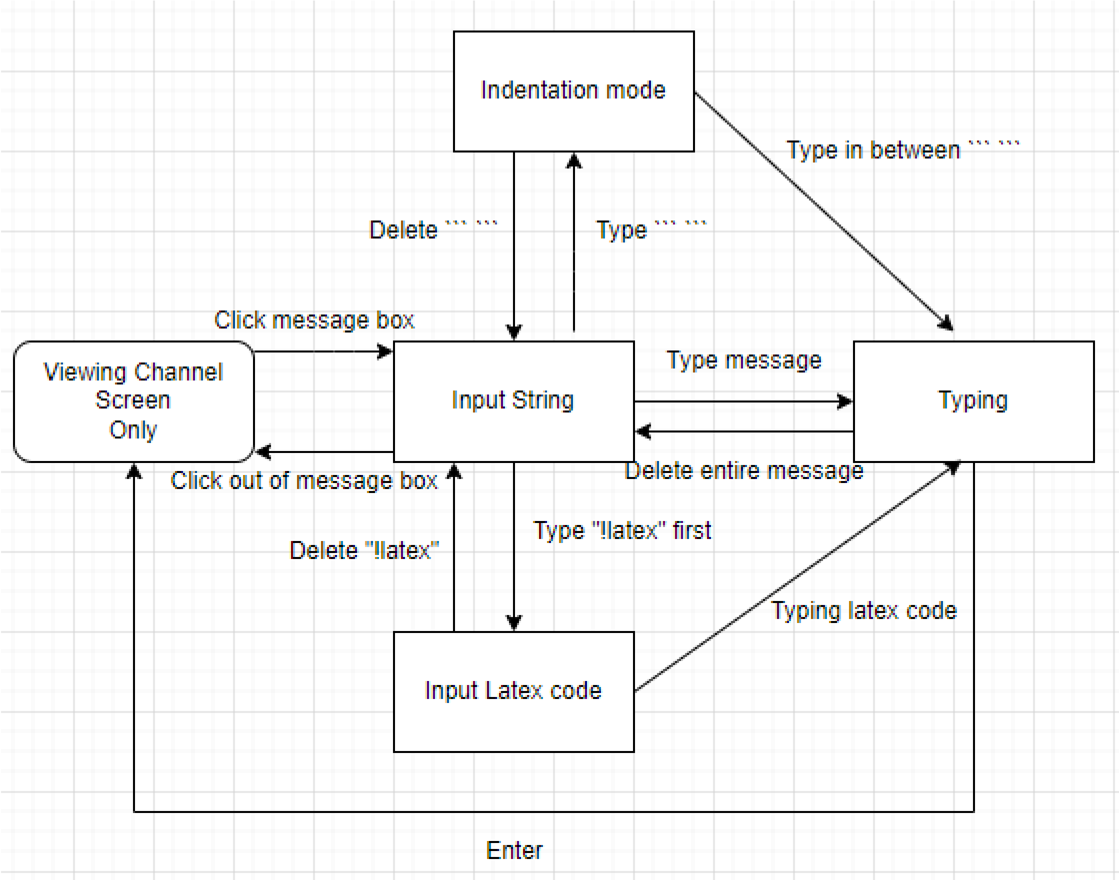

Figure 2: Conceptual model of **message/**

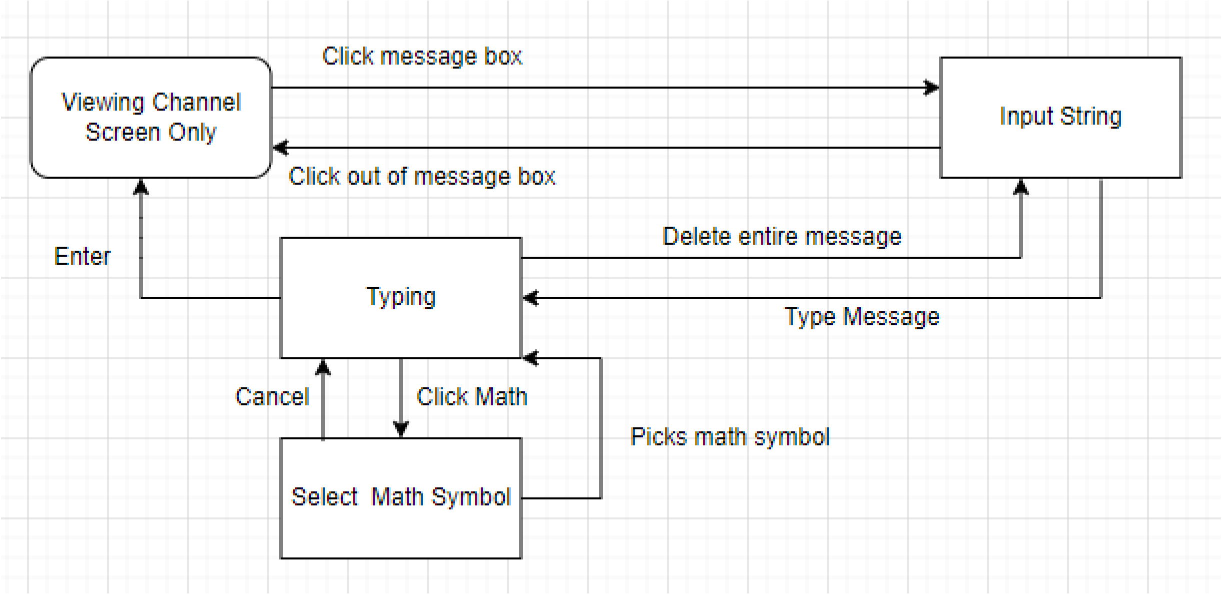

Figure 3: Conceptual model of **maths/send/v1**

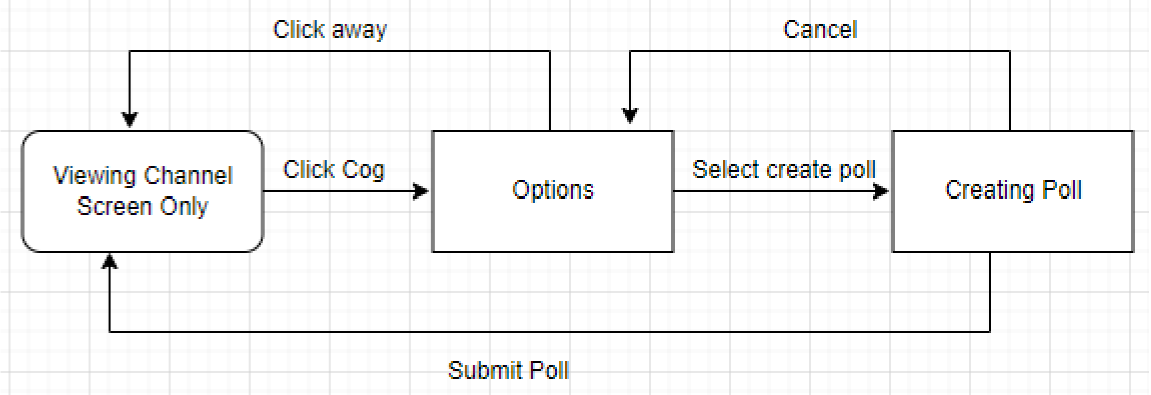

Figure 4: Conceptual model of **poll/send/v1**

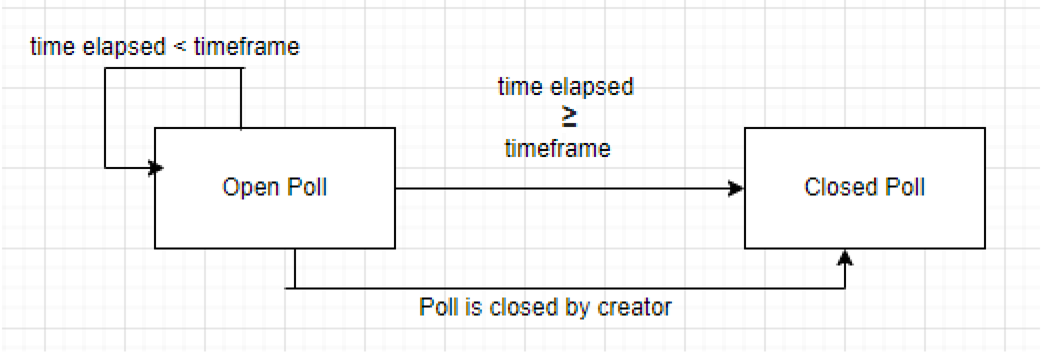

Figure 5: Conceptual model of a poll’s state