

The background features a large, light blue triangle on the left side, pointing towards the top right. The rest of the background is a solid light purple color. The text is centered in the white space between the triangle and the right edge.

Planning For The Next Problems To Solve

Requirements – *Elicitation*

We collected information from 3 target users in the form of a google form survey and asked questions to understand what *problems* they might have with team-work driven communication tools that are currently unsolved by Flockr.

User One

Name

Vivien Tang

Email

Vivnator@gmail.com

What platforms do you use for teamwork oriented projects?

Facebook.

What are some challenges of working in a team?

Communication, finding a time when everyone is free.

What are some good features of good teamwork?

Good communication, everyone does their part.

What is the most important feature you would use for group work?

Different chats for different parts of the assignment.

What features would you like implemented in Flockr to improve team communication?

Reminders for parts being due, a pinboard or something similar where you can see an overview of the task and how it is going.

User Two

Name

Raman Isaac

Email

ramanisaac9677@gmail.com

What platforms do you use for teamwork oriented projects?

Facebook and Discord.

What are some challenges of working in a team?

Communication with the team and planning work.

What are some good features of good teamwork?

Excellent communication and being able to rely on each other to do work to a high standard.

What is the most important feature you would use for group work?
Communication.

What features would you like implemented in Flockr to improve team communication?
Screenshare.

User Three

Name
Bruce Aiken

Email
thomas_aiken@live.com.au

What platforms do you use for teamwork oriented projects?
Discord, Microsoft Teams, Google Drive.

What are some challenges of working in a team?
Getting time to all meet up together.

What are some good features of good teamwork?
Effective communication and time management.

What is the most important feature you would use for group work?
A timetable.

What features would you like implemented in Flockr to improve team communication?
A way private message users within a channel.

Requirements – *Analysis and Specification*

We expressed the requirements from the elicitation as user stories with a user acceptance criteria as notes to have a clear definition of when a story has been completed. We also generated use cases that satisfies the requirements elicited in a written-recipe style.

User One – Vivien Tang

User Story

As a user, I want to be reminded when tasks allocated to me are due, and be able to have access to an overview of the project progress.

Acceptance Criteria – reminders

Scenario: Want to receive a reminder for a task due.

Given: The user is adding a new task to the taskboard.

When: The user ticked the “Send reminder when due” box.

And: The user selects a specific due date for the task.

Then: The system takes the set due date as the date the reminder is sent to the user.

Given: The user enabled the system to send notifications.

When: The user ticks the “Send reminder when due” box.

Then: The system will be timed to send a reminder notification to the user on the due date.

Use case – reminders

Use Case: Reminders for tasks due.

Goal in Context: User receives a reminder notification on the due date set for the task.

Scope: Software, website

Level: Sub-function

Preconditions: The user has been registered, and has joined a channel with other users.

Success End Condition: A reminder notification is sent to the user on the due date set by the user for the task.

Failed End Condition: The user turns off settings for enabling notifications to be received or if the user edits the task details and unticks the “Send reminder” box.

Primary Actor: User

Trigger: User ticks the “Send reminder” box when adding a task to the taskboard.

Steps taken for main success scenario:

1. User registers account on website.
2. User joins channel.

3. User enables notifications in settings.
4. User enters task board.
5. User enters “add a new task”.
6. User selects “public task”.
7. User enters task name and description.
8. User enters progress stage as “to do” or “doing”.
9. User selects a due date for the task.
10. User ticks the “Send reminder” box.
11. User assigns the task to themselves or another user.
12. Task is created and is added to the taskboard.
13. Reminder notification is sent to the user on the due date.

Acceptance criteria – reminders

Scenario: Have access to an overview of the group work progress.

Given: The user(s) have added multiple tasks to the taskboard.

When: The user set the task visibility to be public (can be seen by all members).

And: The user sets specific due dates and progress milestones for each task they add (such as “to do”, “doing”, “completed”).

Then: The system takes the newly added task and posts it on the overview of the taskboard.

When: The user opens the overview of the taskboard on the home page.

Then: The user will have access to an overview of all the tasks that have been created by all the members of the group.

Use case – overview

Use Case: Overview of group progress.

Goal in Context: User is able to view an overview of the task board which incorporates all the public tasks that all users have added.

Scope: Software, website

Level: Sub-function

Preconditions: The user has been registered, and has joined a channel with other users.

Success End Condition: The user is able to view an overview of the task board which displays all the public tasks added by all users and the current stages of the tasks (“to do”, “doing”, “completed”).

Failed End Condition: When the user is no longer in a channel or an authorised user of Flockr.

Primary Actor: User

Trigger: User enters the task board.

Steps taken for main success scenario:

1. User registers account on website.
2. User joins channel.

3. User enters task board.
4. User is able to view all the public tasks that have been added to the task board and their current stages (“to do”, “doing”, “completed”).

User Two – Raman Isaac

User Story

As a user I want to be able to screenshare with my teammates so that we can show each other our progress/work to facilitate teamwork.

Acceptance Criteria – screen share

Scenario: Want to screenshare with groupmates to clarify some points.

Given: The user gave Flockr permission to access their computer screen and audio.

When: The user clicks on “share screen” and decides the screenshare preferences.

And: The groupmates click on “view <user>’s shared screen”.

Then: The group mates are able to see and hear the contents of the host’s screen.

When: The user clicks on “stop sharing screen”.

Then: No longer can anyone view or hear the contents of the user’s screen.

Use case – screen share

Use Case: Share screen with other members.

Goal in Context: Users can share the contents (audio and visual) of their screen with other members in the channel.

Scope: Software, website

Level: Sub-function

Preconditions: The user has been registered, has joined a channel with other users and allowed the system to access and share the user’s screen and audio output.

Success End Condition: The host is able to share the contents of their screen with the other members who joined the stream.

Failed End Condition: The contents of the screen could not be shared with other members despite clicking on “share screen” and giving permission for the system to access the relevant parts of the computer.

Primary Actor: User

Trigger: User clicks the “Share screen” button in the chat channel.

Steps taken for main success scenario:

1. User registers account on website
2. User joins and enters channel with groupmates
3. User clicks on “share screen” and gives permission to the system to access and share the host’s screen and audio output.
4. Channel members are able to join the stream and see/hear what the host is able to.

User Three – Bruce Aiken

User Story

As a user I want to be able to private message users within a channel so that I am able to effectively communicate with a teammate one-on-one and privately.

Acceptance Criteria – private message

Given the user has navigated to the channel they are a member of

When the user views the list of users within the channel

And clicks on the desired user they wish to privately message

Then the user is redirected to a list of options with private messaging being one

And the user clicks on the private messaging option

Then the user is redirected to a chatbox with the desired user

Use case – private message

Use Case: Private message

Goal in Context: Users privately send a message to a user within a channel they're a member of

Scope: Software, website

Level: Sub-function

Preconditions: The user has been registered, and has joined a channel with other users.

Success End Condition: The message has been sent to the desired user

Failed End Condition: The message has not been sent to the desired user

Primary Actor: User

Trigger: User clicks on the private message option with a desired user within a channel.

Steps taken for main success scenario:

1. User registers account on website
2. User joins channel
3. User enters channel
4. User selects targeted user
5. User selects private message option from list
6. User types in message and presses send
7. Message is uploaded to the private chat between users.

Requirements – *Validation*

We inquired our target users about the extent to which the use cases would adequately describe the problem they're trying to solve. Here are their comments.

User One – Vivien Tang

“Overall, the use case for a task board overview addressed the issue quite well. However for reminders, one little thing that could be added is the option for constant reminders for those who procrastinate or forget, otherwise it addresses the issue extensively.”

User Two – Raman Isaac

“I rated the solution 10/10. The use case addressed all of the needs that I could think of in a simple manner. That is all I would desire from a screen share function for effective communication”

User Three – Bruce Aiken

“The use case achieved my requirements and completed what I had asked for so the extent to which it described the problem I wanted to solve with Flockr was pretty great”

Design – *Design Interface*

We have specified the capabilities necessary as HTTP endpoints in consideration of *solutions* to our problems.

Reminders

Route	taskboard/send_reminders
Method	POST
Parameters	(taskboard_id, token, u_id, task_id, reminder_id, notification_permission)
Return Type	{reminder_id}
Exceptions	InputError: <ul style="list-style-type: none">• Invalid taskboard_id.• Invalid token.• Invalid u_id.• Invalid task_id. AccessError: <ul style="list-style-type: none">• When notifications are turned off → permissions are denied.• When a user that did not create the task attempts to tick “Send reminder” when editing a task from the task board.• When the task has been deleted.
Description	Sends a reminder notification to the user that the task created has been assigned to.

Route	taskboard/delete_reminders
Method	DELETE
Parameters	(taskboard_id, token, u_id, task_id, reminder_id, notification_permission)
Return Type	{}
Exceptions	InputError: <ul style="list-style-type: none">• Invalid taskboard_id.• Invalid token.• Invalid u_id.• Invalid task_id.

	AccessError: <ul style="list-style-type: none"> When a user that did not create the task attempts to untick “Send reminder” when editing a task from the task board. When the task has been deleted.
Description	Deletes a reminder notification that was scheduled to be sent to the user that the task created has been assigned to.

Overview

Route	taskboard/overview
Method	POST
Parameters	(taskboard_id, token, task_id)
Return Type	{tasks_list}
Exceptions	InputError: <ul style="list-style-type: none"> Invalid taskboard_id. Invalid token. Invalid task_id. Private tasks cannot be added to the task board overview for public display. AccessError: <ul style="list-style-type: none"> When the user is not part of the Flockr or not in any relevant channels.
Description	Displays an overview of all the public tasks that have been added to the task board.

Screenshare

Route	channel/start_sharescreen
Method	POST
Parameters	(channel_id, token, access_screen_permission, access_audio_permission)
Return Type	{}
Exceptions	Input error: <ul style="list-style-type: none"> Invalid channel_id. Invalid token.

	<ul style="list-style-type: none"> access_screen_permission and access_audio_permission denied.
Description	Start sharing screen.

Route	channel/edit_sharescreen
Method	PUT
Parameters	(sharescreen_id, token, access_screen_permission, access_audio_permission)
Return Type	{}
Exceptions	Input error: <ul style="list-style-type: none"> Invalid sharescreen_id. access_screen_permission and access_audio_permission denied. Access error: <ul style="list-style-type: none"> Invalid token (caller is not host).
Description	Edit the options of screen share function.

Route	channel/stop_sharescreen
Method	POST
Parameters	(sharescreen_id, token)
Return Type	{}
Exceptions	Input error: <ul style="list-style-type: none"> Invalid sharescreen_id. Access error: <ul style="list-style-type: none"> Invalid token (caller is not host).
Description	Stop sharing screen.

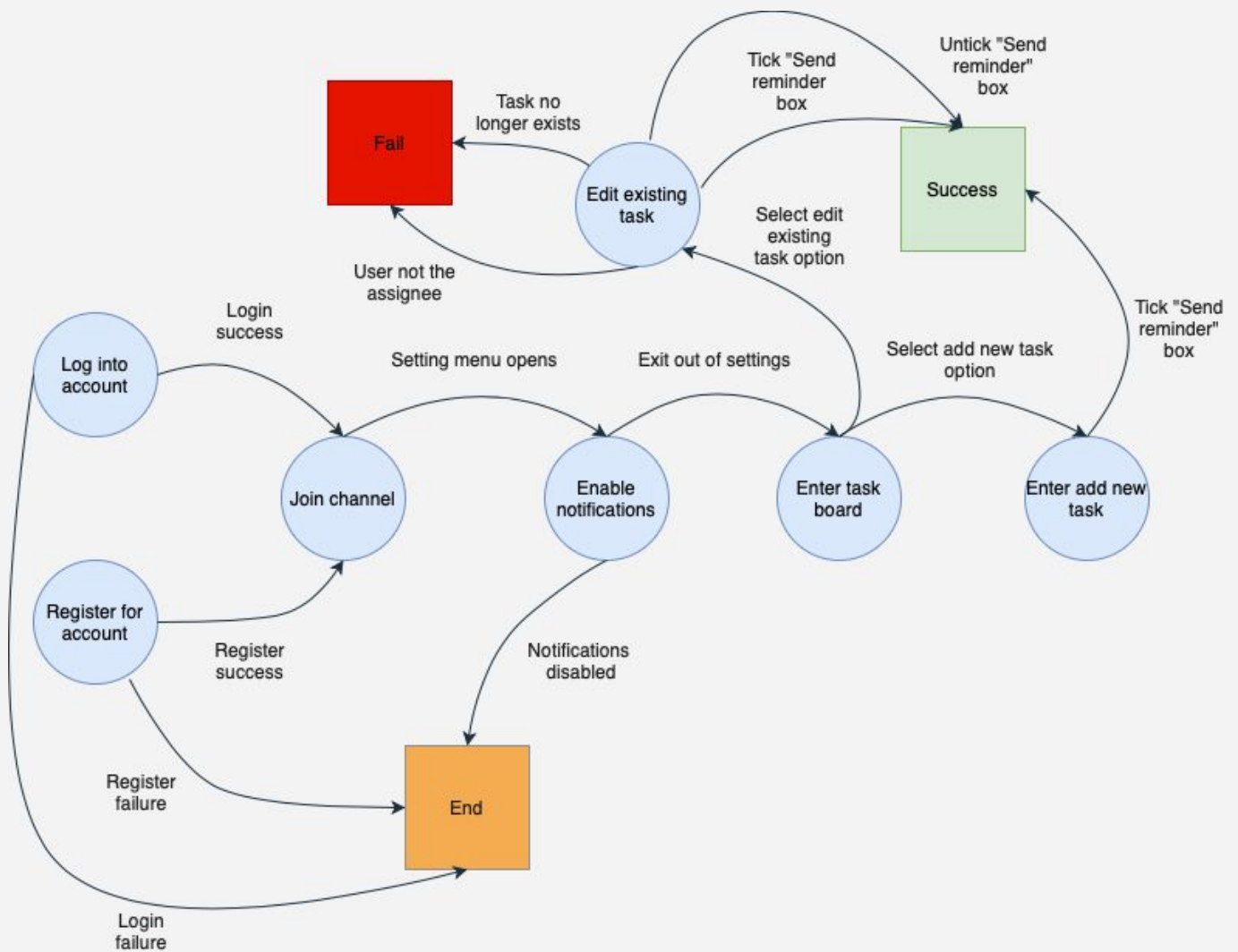
Private message

Route	message/private_send
Method	POST
Parameters	(token, user_id, message)
Return Type	{message_id}
Exceptions	<p>InputError when:</p> <ul style="list-style-type: none"> • Message is more than 1000 characters. <p>AccessError when:</p> <ul style="list-style-type: none"> • The authorised user has not joined the channel the user they're trying to send a message to is in.
Description	Send a message from authorised_user to the user specified by u_id.

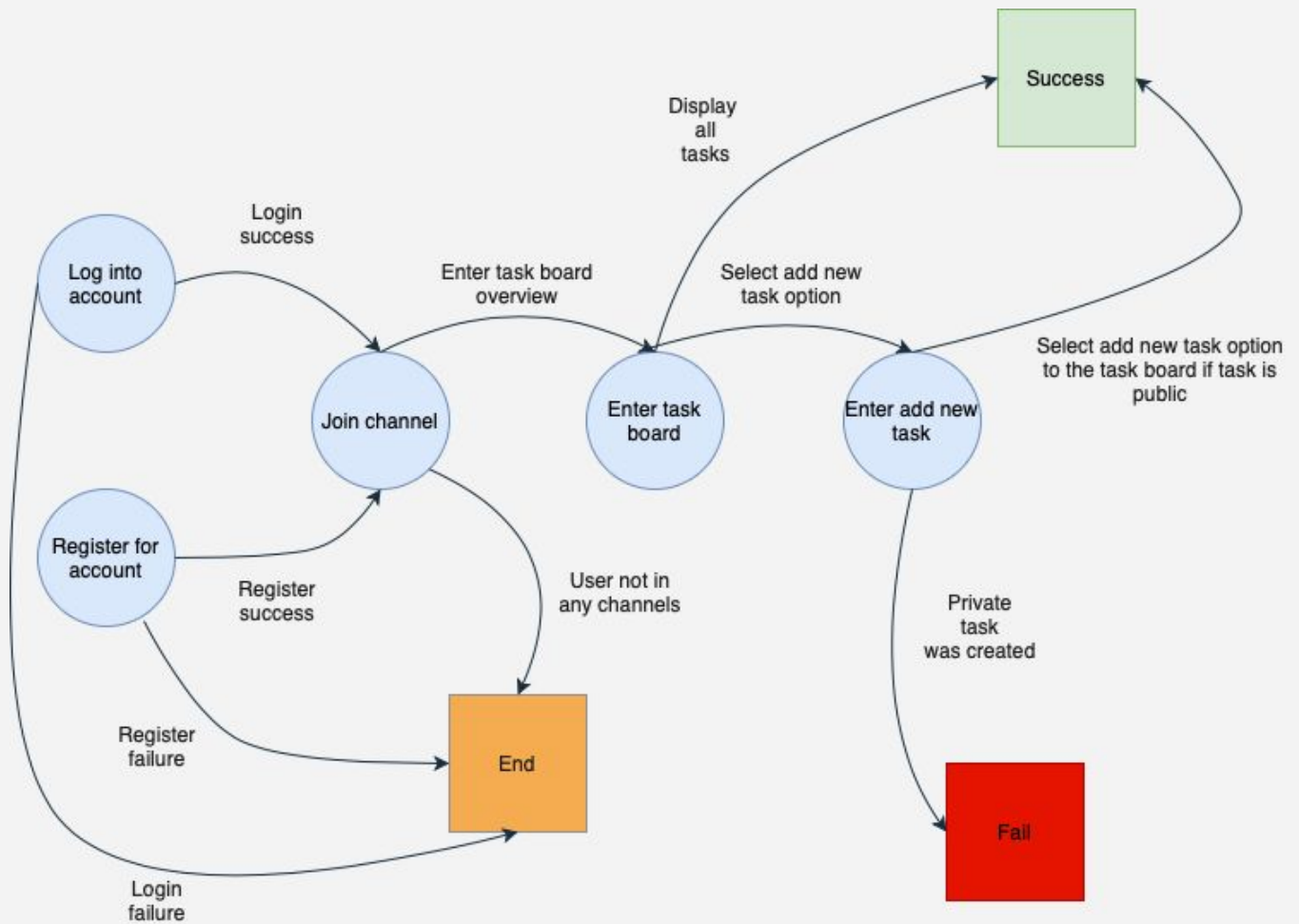
Design - *Conceptual Modelling (state)*

We have created state diagrams in order to show how the state of application would change based on user actions.

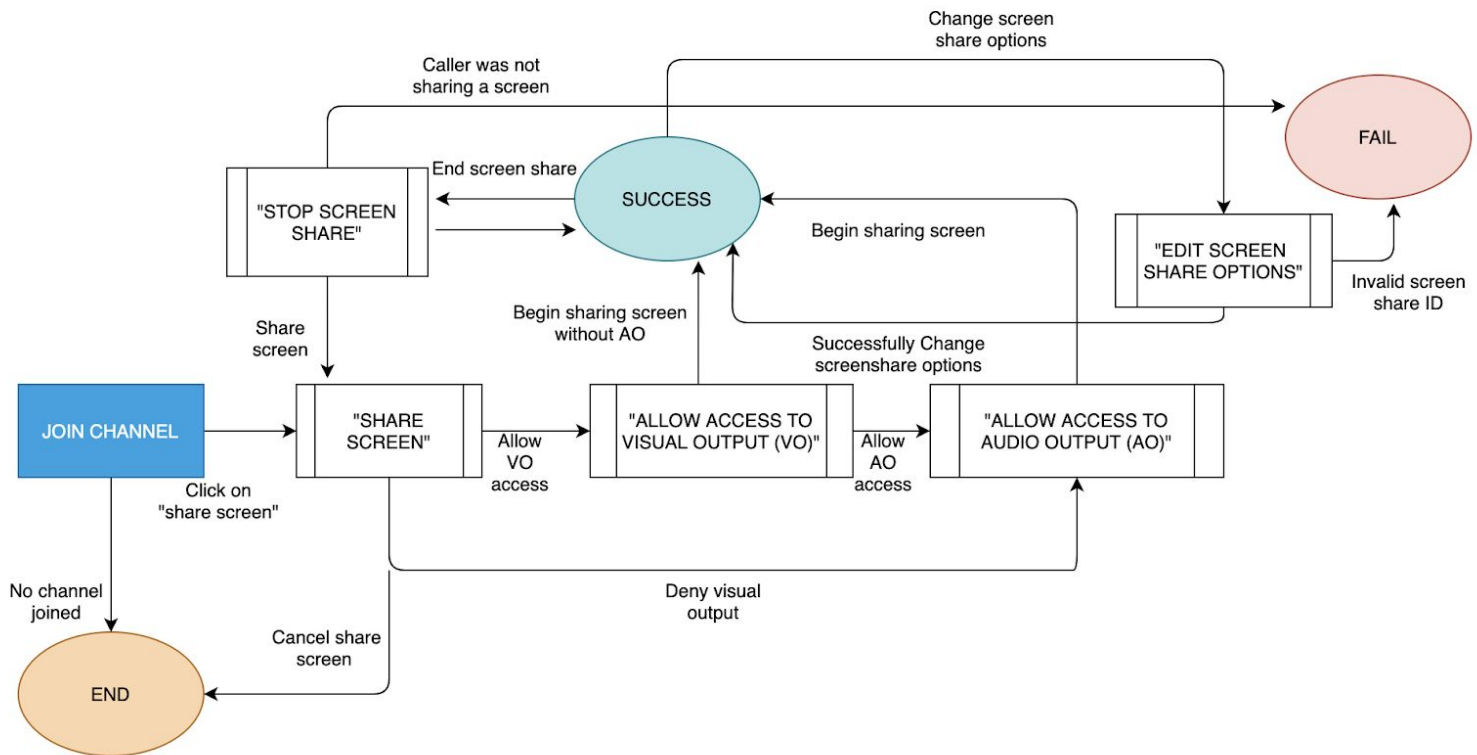
Reminders



Overview



Screenshare



Private message

