

H15A_DREAM Iteration 3 Report

0 Table of Contents

[0 Table of Contents](#)

[1 Elicitation](#)

[1.1 Questions](#)

[1.2 Interviews](#)

[user 1](#)

[user 2](#)

[user 3](#)

[2 Analysis & Specification — Use Cases](#)

[2.1 User Stories](#)

[channel calendar](#)

[upvote / downvote messages](#)

[2.2 Use Cases](#)

[channel calendar](#)

[upvote / downvote messages](#)

[3 Validation](#)

[4 Interface Design](#)

[channel calendar](#)

[upvote / downvote messages + sort by votes](#)

[5 State Diagrams](#)

[add event to calendar / remove event from calendar use case](#)

[sort by votes use case](#)

1 Elicitation

1.1 Questions

1. Are there any aspects that you are currently dissatisfied with in any teamwork-driven communication tools that you are using?
 - How would you ideally like to see these problems get resolved?
2. When deciding on which communication tool to use for a teamwork environment what factors do you consider?
 - What would encourage you to pick UNSW Treats over other tools?
3. What features would you like to see added to a communication tool such as UNSW Treats?
4. What's the feature you find you use most frequently in your current communication tools?

1.2 Interviews

user 1

name: Issac Liu

email: issac.liu.il.03@gmail.com

responses

1. **Are there any aspects that you are currently dissatisfied with in any teamwork-driven communication tools that you are using? How would you ideally like to see these problems get resolved?**

I currently use Microsoft teams for team-work driven communication. I'm not really dissatisfied with any aspects or features within teams.

2. **When deciding on which communication tool to use for a teamwork environment what factors do you consider? What would encourage you to pick UNSW Treats over other tools?**

Organisational features are essential when I'm deciding between communication applications. Tools that improve organisation are tools that would influence me to choose treats over other applications.

3. **What features would you like to see added to a communication tool such as UNSW Treats?**

I find calendar features on teams extremely useful for managing and organising due dates, events, etc.

I also find that splitting channels into further subchannels organises different conversations about different topics, and prevents high frequencies of conversations that are irrelevant to each other.

4. **What's the feature you find you use most frequently in your current communication tools?**

I used the standard tools of messaging to different subchannels and updating a public channel calendar quite often.

problem UNSW Treats lacks any organising tools that are essential for teamwork-driven communication.

possible solution Implement a calendar feature along with subchannels within channels.

user 2

name: Mondira Paul

email: mondirapaul26@gmail.com

responses

1. **Are there any aspects that you are currently dissatisfied with in any teamwork-driven communication tools that you are using? How would you ideally like to see these problems get resolved?**

I mostly use tools like microsoft teams for education and online learning. Something I find annoying is the lack of utility that current engagement features have — e.g. reacting with emojis is nice, but not very helpful to a presenter or tutor.

2. **When deciding on which communication tool to use for a teamwork environment what factors do you consider? What would encourage you to pick UNSW Treats over other tools?**

Something that I strongly consider is the way in which an app facilitates engagement and user responses, whether that's through emoji reactions, message pins and polling/voting systems.

3. **What features would you like to see added to a communication tool such as UNSW Treats?**

As someone who would use an app like Treats for educational purposes, I have issues with how easy it is to become lost in an overflow of messages / notifications. Because of this, I would really like to see a voting system where users can vote for messages in channels that are most relevant / important (e.g. someone asks a question that everyone else wants to know the answer to)

4. **What's the feature you find you use most frequently in your current communication tools?**

I mainly use MS Teams for educational purposes – (med faculty runs all lectures etc. through teams). Thus, I most frequently use messaging features and chat rooms as a way to ask questions etc for my learning.

problem UNSW Treats lacks sufficient features for user engagement, especially for education as a context in which it could be used.

possible solution Implement a voting system for pinned messages so that users can see content (e.g. questions or messages or announcements) that is more relevant (and facilitate higher engagement)

user 3

name: Jemima Owen

email: jemima.owen@gmail.com

responses

1. **Are there any aspects that you are currently dissatisfied with in any teamwork-driven communication tools that you are using? How would you ideally like to see these problems get resolved?**

I currently use MS as a teamwork-driven communication tool. One inconvenient thing is that when I navigate the files and select a file, there isn't an option to drive me to the chat history where this file

is located.

2. **When deciding on which communication tool to use for a teamwork environment what factors do you consider? What would encourage you to pick UNSW Treats over other tools?**

The site is simple and clear to use which would not distract my attention when I'm doing some important stuff.

3. **What features would you like to see added to a communication tool such as UNSW Treats?**

A photo album would be useful for navigating and organising important pictures sent by group members.

4. **What's the feature you find you use most frequently in your current communication tools?**

I like the message edit feature, it's quite convenient when there is a typo sometimes or some small modification compared to the previous to do, rather than having to send a new unnecessary message.

problem UNSW treats lack of a to-do board, such as the feature similar to Trello, which might be helpful to organise projects.

possible solution Implemented a to-do board feature for every channel and user.

2 Analysis & Specification — Use Cases

2.1 User Stories

channel calendar

user stories

as a UNSW treats user I want a weekly public channel calendar, to facilitate the needs to organise events and specify important dates

as a UNSW treats user I want channel owners to be able to add events to the channel calendar with details like the event name, a brief description summarising the event's contents, the starting time and ending time of the event

as a UNSW treats user I want to be able to remove existing events from the calendar because the event has been cancelled/postponed

user acceptance criteria

- calendars should display dates starting from Monday to Sunday, in which the current date lies in between
- when a channel owner clicks on the add event button they will be prompted with a form requiring a name, date, description, starting time and ending time for the event
- when a channel owner adds an event to the calendar a tag should appear on the public channel calendar on the specified date stating the name, starting time and ending time of the event
- when a channel owner clicks on the remove event button the event tag will disappear from the public calendar
- when users click on an event tag the name, date, description, starting time and ending time of the event will be displayed
- users that are only channel members cannot add events to the public calendar
- users that are only channel members cannot remove existing events on the public calendar

upvote / downvote messages

user story	<p>as a UNSW treats user (member of channel / dm), I want to be able to upvote or downvote pinned messages, so that messages that are particularly important or relevant are more prominent</p> <p>as a UNSW treats user, I want to be able to sort the pinned messages in a channel based on their votes, so that I can which messages matter most / are most relevant to me</p>
user acceptance criteria	<ul style="list-style-type: none"> • each pinned message should have a upvote and downvote button • each pinned message should have a counter keeping track of the votes • when the user clicks on the upvote button the counter will increase by 1 • when the user clicks on the downvote button the counter will decrease by 1 • users cannot vote twice for the same message, but can change their vote • there will be an checkbox that allows the user to view the pinned messages in descending order of votes, rather than in terms of most recent send-time

2.2 Use Cases

channel calendar

use case 1: add events to calendar

Goal in Context: users can add events to a channel calendar

Scope: within a channel that the user has joined

Preconditions: user must be a channel owner

Success End Condition: event tag added to the calendar

Failed End Condition: no event is added to the calendar

Primary Actor: a user that is a channel member within the channel

Trigger: user fills a form requiring event details and clicks on 'add event' button

use case 1 steps

Step 1: user clicks on 'calendar' button within the channel's page

Step 2: system displays a weekly calendar

Step 3: user clicks on 'add event button'

Step 4: system displays a form requiring event information

Step 5: user fills the event details (name, date, description, starting time, ending time)

Step 6: user clicks on 'submit' button

Step 7: system display refreshed calendar with event tag on the corresponding date

use case 2: remove events from calendar

Goal in Context: users remove existing events from the channel calendar

Scope: within a channel that the user has joined

Preconditions: user must be a channel owner

Success End Condition: event tag removed from the calendar

Failed End Condition: event is still on the calendar

Primary Actor: a user that is a channel member within the channel

Trigger: user clicks on 'remove event' button

use case 2 steps

Step 1: user clicks on 'calendar' button within the channel's page

- Step 2:** system displays a weekly calendar
- Step 3:** user clicks on the event tag on the calendar
- Step 4:** system displays event information
- Step 5:** user clicks on 'remove event' button
- Step 6:** system displays refreshed calendar with the event tag removed
-

upvote / downvote messages

use case 1: upvote / downvote a message

- Goal in Context:** the user upvotes / downvotes a message
- Scope:** within a channel / dm that the user has joined
- Preconditions:** a message sent in a channel / dm is pinned
- Success End Condition:** message vote count updated
- Failed End Condition:** no change
- Primary Actor:** user that is a member of a channel / dm
- Trigger:** user clicks the relevant upvote / downvote button

use case 1 steps

- Step 1:** when a message is pinned, upvote / downvote buttons appear for user to click + system updates the message's "votes counter"
- Step 2:** when the user clicks the upvote / downvote button, the message "votes counter" changes appropriately (+1 for upvote, -1 for downvote)
- Step 3:** the system updates (the database) accordingly such that all relevant users can see the changes
- Step 4:** if the 'sort by votes' checkbox is selected, the system reorders the pinned messages in descending order by votes, according to the newly updated votes
-

use case 2: sort by votes

- Goal in Context:** the user can view the pinned messages in order
- Scope:** within a channel / dm that the user has joined
- Preconditions:** messages sent in a channel / dm are pinned and have been voted for
- Success End Condition:** pinned messages sorted by votes
- Failed End Condition:** pinned messages sorted by time sent
- Primary Actor:** user that is a member of a channel / dm
- Trigger:** user clicks the 'sort by votes' checkbox

use case 2 steps

- Step 1:** system displays channel / dm pinned messages and their vote counts
- Step 2:** user clicks 'sort by votes' checkbox
- Step 3:** system sorts pinned messages in descending order of votes, then by time sent
- Step 4:** resulting message order is displayed to the user
-

3 Validation

use case	feedback
channel calendar	I think these use cases would be an effective organisation tool that would solve a lot of miscommunications with important dates within a team-driven environment.

use case	feedback
upvote / downvote messages	These use cases seem to effectively solve the issue and would be something I would like to see.
sort by votes	A possible idea is to have more customisability and user settings to do these things by default instead of requiring a checkbox to trigger, but otherwise a very helpful idea.

4 Interface Design

channel calendar

input / output types

variable name	input/output type
has suffix <code>Id</code>	<code>number</code> — refers to the unique id of that element
<code>event</code>	<code>Array</code> of objects with types { <code>eventId</code> , <code>eventName</code> , <code>eventDescription</code> , <code>eventTimeStart</code> , <code>eventTimeEnd</code> , <code>eventDate</code> }
<code>eventName</code>	<code>string</code> — the name of the event
<code>eventDescription</code>	<code>string</code> — brief description about the event
<code>eventTimeStart</code>	<code>string</code> - a 24 hour time format (hh.mm)
<code>eventTimeEnd</code>	<code>string</code> - a 24 hour time format (hh.mm)
<code>eventDate</code>	<code>string</code> - a date format (dd/mm/yyyy)
<code>currentDate</code>	<code>string</code> - a date format (dd/mm/yyyy)
<code>eventInfo</code>	<code>Object</code> containing keys <code>eventName</code> , <code>eventDescription</code> , <code>eventDate</code> , <code>eventTimeStart</code> , <code>eventTimeEnd</code>

all routes take `token` in the request header and return `403 ERROR` if the `token` is invalid

name and description	HTTP Method	Data Types	Exceptions
<code>calendar/event/v1</code> given channelId and the current date returns the name, date and time of all events during a week for the specified channels	GET	Query Parameters: { <code>channelId</code> , <code>currentDate</code> } Returns: { <code>eventInfo</code> }	400 Error: invalid channelId; user is not in the channel
<code>calendar/eventAdd/v1</code> given information regarding a new event, adds a new event to the channel calendar	POST	Body Parameters: { <code>channelId</code> , <code>eventName</code> , <code>eventDescription</code> , <code>eventDate</code> , <code>eventTimeStart</code> , <code>eventTimeEnd</code> } Returns: { <code>eventId</code> }	400 Error: invalid channelId; user is not in the channel; <code>eventTimeStart</code> is later than <code>eventTimeEnd</code> ; <code>eventName</code> is an empty string; <code>eventDate</code> is an empty string; 403 Error: user is in the channel but not a channel owner
<code>calendar/eventRemove/v1</code> given a eventId and channelId it removes the corresponding event from the calendar	POST	Body Parameters: { <code>channelId</code> , <code>eventId</code> } Returns: { }	400 Error: <code>channelId</code> is invalid; <code>eventId</code> is invalid; 403 Error: user is in the channel but not a channel owner

upvote / downvote messages + sort by votes

input / output types

variable name	input/output type
has suffix <code>Id</code>	<code>number</code> — refers to the unique id of that element
<code>voteCount</code>	<code>number</code> — result when totalling all upvotes / downvotes
<code>userVote</code>	<code>number</code> — <code>-1</code> = user has downvoted, <code>0</code> = user has not voted, <code>1</code> = user has upvoted
<code>pinnedMessages</code>	<code>Array</code> of objects with types { <code>messageId</code> , <code>uId</code> , <code>message</code> , <code>timeSent</code> , <code>reacts</code> , <code>voteCount</code> , <code>userVote</code> }
<code>message</code>	<code>string</code> — the contents of the message
<code>timeSent</code>	<code>number</code> — unix timestamp in seconds
<code>reacts</code>	<code>Array</code> of objects with types { <code>reactId</code> , <code>uIds</code> , <code>isThisUserReacted</code> }

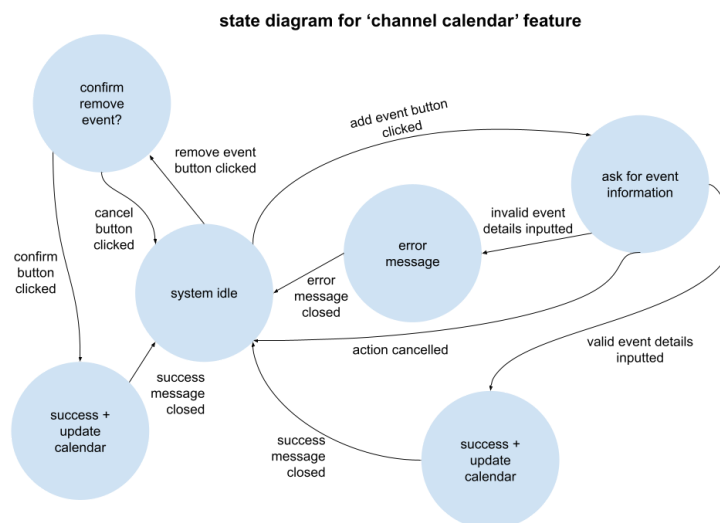
variable name	input/output type
<code>uIds</code>	Array of <code>uid</code> elements — lists users who have reacted to the message
<code>isThisUserReacted</code>	boolean — whether the user who called the route has reacted to the message with that react

all routes take `token` in the request header and return `403 ERROR` if the `token` is invalid

name and description	HTTP Method	Data Types	Exceptions
<code>message/countvotes/v1</code> given a message, returns the message's vote count and the vote of the current user	GET	Query Parameters: { <code>messageId</code> } Success return type: { <code>voteCount</code> , <code>userVote</code> }	400 Error: - invalid <code>messageId</code> (i.e., message does not exist) - user is not in the same channel/dm as message - message is not pinned
<code>message/upvote/v1</code> allows a user to upvote a message; if the user previously downvoted the message, their vote is changed to an upvote	PUT	Body Parameters: { <code>messageId</code> } Success return type: {}	400 Error: - invalid <code>messageId</code> (i.e., message does not exist) - user is not in the same channel/dm as message - message is not pinned - the user has already upvoted the message
<code>message/downvote/v1</code> allows a user to downvote a message; if the user previously upvoted the message, their vote is changed to a downvote	PUT	Body Parameters: { <code>messageId</code> } Success return type: {}	400 Error: - invalid <code>messageId</code> (i.e., message does not exist) - user is not in the same channel/dm as message - message is not pinned - the user has already downvoted the message
<code>channel/messages/votes/v1</code> gets the pinned messages of a channel sorted in descending order by votes, then by time sent	GET	Query Parameters: { <code>channelId</code> } Success return type: { <code>pinnedMessages</code> }	400 Error: - invalid <code>channelId</code> (i.e., channel does not exist) 403 Error: - valid <code>channelId</code> but user is not part of channel
<code>dm/messages/votes/v1</code> gets the pinned messages of a dm sorted in descending order by votes, then by time sent	GET	Query Parameters: { <code>dmId</code> } Success return type: { <code>pinnedMessages</code> }	400 Error: - invalid <code>dmId</code> (i.e., dm does not exist) 403 Error: - valid <code>dmId</code> but user is not part of dm

5 State Diagrams

add event to calendar / remove event from calendar use case



sort by votes use case

state diagram for 'sort by votes' feature

