T11B_BOOST

This document first provides a brief functionality report of the current iteration of the workplace messaging platform, Beans. After interviewing 3 potential users, we have identified some key features that are important to the target market that could be implemented in future iterations. One function from the interviews is explored further.

The current functionality of Beans:

Admin	Auth	Channel	Channels	Dm	Message	Search	Standup	Users	Notifications
Remove a user Change a user's permissions	Register an account Login Logout Reset a password Request an email to reset password	Obtain brief details of the channel (channel name & ID number) Join a channel Invite a user to a channel List out all messages within a channel Leave a channel Add an owner to a channel Remove an owner of a channel	Create a channel (public or private) List the channels the user is apart of List all the channels in Beans	Create a DM between yourself and a group of different users List the DMs that the user is apart of Remove a dm Obtain brief details about a DM Leave a DM List all messages within a DM Send a DM message at a later time	Send a message in a channel Edit a message in a dm or channel Remove a message Share a message React to a message Unreact to a message Pin a message Unpin a message Send a message at a later time	Search all channels & DMs a user is apart of for a query substring	Begin a standup (meeting) for a given amount of time Obtain information about whether the standup is still active Send a message within a standup	Obtain the profile of a user List all users registered in Beans Set a profile name Set an email Set a handle Upload a profile photo Fetch statistics of a user Fetch statistics of the entire workspace's use of Beans	Return a list of notifications for the user

Interview Questions

1. What platform do you currently use at work & what do you think is the best feature that it has?

2. Given the current functionality of Beans, what do you think is lacking and if implemented would improve your teamwork-driven communication?

3. Why would this functionality help you & what benefit does it bring?

4. Is there any specific behaviour you would like to see this function have?

Interview 1:

Megan Pozzi, megan.pozzi@qut.edu.au, Manager of Language and Learning at QUT

1. What platform do you currently use at work & what do you think is the best feature that it has?

   "So we use Microsoft Teams and I think that the best feature is it's integration with outlook so that when a meeting is scheduled on Teams it will then go straight into my calendar on outlook. It's also just a better, more informal way to talk to my colleagues"

2. Given the current functionality of Beans, what do you think is lacking and if implemented would improve your teamwork-driven communication?

   "Yeah just looking at it's current functionality, I think it would be really important to add a status function to indicate that users are busy or away so that they shouldn't or can't be contacted depending on their status"

3. Why would this functionality help you/what benefit does it bring?

"Adding in a status function would just not only help to not create distractions when I'm working on a task or in a current meeting, but it's also just good for people's mental health, like I don't want to be contacted when I'm having lunch or when I'm taking the day off!"

4. Is there any specific behaviour you would like to see this function have?

"Specifically I think it should be like if another person tries to contact me when I have a busy or away status set, they should then be notified back that the message won't be sent to me until I'm available again, because obviously people can still send me messages but a notification back to the person sending me the message will deter them from sending anything else or will make them only message me when it's important"

Proposed Solution to Interview 1

Megan has identified her desire for a status indicator within the platform so that her colleagues are deterred from messaging her when her status bar is not set to available. This could be achieved in the Beans platform through a set status bar function for each user that can set their status to simply "available" or to "away". Then if a user A sends a direct message or is tagging user B, who has an away status in a chat, then user A is then notified that user B is away and will not receive the message until they are available again.

Interview 2:

Ram Page, rampage202@proton.me, Case Management Specialist @ EML

1. What platform do you currently use at work & what do you think is the best feature that it has?

"We use Guidewire and actually nothing is good about it, it's so slow and I hate it but if i had to say something good, it saves us time by auto-generating common letter templates, because we send out a lot of emails to clients so there is a directory on there that allows us to send out letters to people super quickly"

2. Given the current functionality of Beans, what do you think is lacking and if implemented would improve your teamwork-driven communication?

"I guess this would be adding onto my point before but in a more simple way, it'd be good if there were some auto-generated responses to common phrases that I could send to colleagues in a chat"

3. Why would this functionality help you & what benefit does it bring?

"This would save me a lot of time and energy from thinking of a response and typing it out, it also means that there is a shared appreciation for the responses especially if it's an informal message being sent, like my colleagues just can expect me to acknowledge the message and more on"

4. Is there any specific behaviour you would like to see this function have?

"What I'm thinking is that when a message is sent in a group or channel and particularly if I am tagged in the message, so then I know the message is targeted toward me, then there should be suggested responses returned back to me that I can send back to the person, so like for example if someone writes 'Ram, can you take a look at this specific case', the platform should then return back to me some common replies like 'sure, no worries', 'I won't be able to do that today' or 'already done'

Proposed Solution to Interview 2

Ram has indicated that he wishes Beans had functionality to generate common reply types so that he can quickly respond to a message in a channel. This could be achieved in Beans by storing common key message types like 'can you…?', 'please take note of…' and 'there will be a [something] at [time]...', then Beans can return appropriate responses back to the user like 'yes', 'no', 'sure, take your time!' etc. the user will then select a choice from the suggestions or can write their own reply.

Interview 3:

Jess King, jessking2031@gmail.com, Graduate Psychology Student @ Unimelb
1. What platform do you currently use at work & what do you think is the best feature that it has?
   "When I'm working on a project at university, we most commonly use discord to chat and call, and I think that the best feature is that I can remove or edit messages because we often miss say something and need to fix it or delete it entirely"

2. Given the current functionality of Beans, what do you think is lacking and if implemented would improve your teamwork-driven communication?

"*Actually I think something that would be good is a report button for inappropriate messages because sometimes maybe by accident or like someone is having a breakdown in your team, people can send really rude or inappropriate messages and those messages need to be reported and taken down, I'm actually saying this because it happened to me last week where we had to report back to our course coordinator that someone in the group was sending really bad messages*"

3. Why would this functionality help you & what benefit does it bring?
   "*Well I think it ensures that the chat stays on topic and means that no one is going to be hurt or offended by any messages that are sent*"

4. Is there any specific behaviour you would like to see this function have?
   "*Yeah so really someone should be able to report a message, and probably too often this happens accidentally so like it's probably better that if someone reports a message, it should then be sent to like the admin of the group to be reviewed then they can make the decision whether or not the message stays in the group or is deleted*"

Proposed Solution to Interview 3

Jess has commented that recently there has been problems in some chats that she is in where users are sending inappropriate messages that should be taken down. This problem can be addressed in Beans if there were a functionality available to report messages, the message can then be sent to owner members of the channel/DM to either be deleted from the channel or deemed appropriate and the message will remain.

User Stories

As a manager and user of the chats, I don't want to be notified of action taking place on the platform when I set a particular status so that I can focus when I'm busy or enjoy my time off when I'm not working.

Acceptance Criteria:
- The user can set availability statuses to either available or away.
- If the user is 'available', the actions that other users make that create notifications will be generated and stored as normal.
- If the user is 'away', any actions that create notifications directed toward the user will still be generated and stored for later, however the user creating the notifications will then be notified that the user is away and will not be receiving notifications until their status is set back to available so they should not be contacted.

VALIDATION:

"For the most part, I think you have captured exactly what I was trying to explain, I like it that people are notified back that I'm away, so they are prompted to stop sending notifications to me. I wonder if you could also set a 'busy' status that could push me the notifications after a certain time block"
- *Megan Pozzi*

As a user, I want to be able to use quick replies to messages that don't require elaboration so that I can save time and energy replying to common messages.

Acceptance Criteria:
- Once a user sends a message that contains a common phrase, appropriate reply phrases should be generated based on the message content and returned to users in the channel or DM.
- Users can choose to respond to that message by choosing one of the suggested replies or can type their own message.

"I think this is great, I especially like that I can also just write a message too, hopefully in the future you can build a database of common phrases so that the replies can become more accurate"
- *Ram Page*

As a user, I want to be able to report inappropriate messages so that the chat can stay on topic and not be offensive in any way.

Acceptance Criteria:
- If a user sends a message that another user deems to be inappropriate, the user can report the message.
- Once the message has be reported, the message content will be changed to 'message has been reported'
- Once reported, the message, in its original form, is sent to the owner members of the channel to be approved or denied.
- Only one owner member needs to approve or deny the message.
- If the message is approved, the message is changed back to the original message, as it was not inappropriate for the chat
- If the message was denied and deemed inappropriate, the message content will be removed entirely.

Use Case: Reporting Messages

1. Users register and login to Beans.
2. A channel or DM is created.
3. A number of authorised users join or are invited to the chat. The chat includes at least one owner member.
4. Members within the chat send messages.
5. User A sends an inappropriate message that offends user B.
6. User B reports the message as inappropriate.
7. The message content is changed to be 'Message has been reported'.
8. All of the owner members of the chat are notified that there has been an inappropriate message flagged.

9. The message, in its original form, is reviewed by only one owner member.
10. The owner member either deems the message suitable, and the message is then changed back to its original form in the chat, or it is deemed inappropriate and is deleted from the chat.
11. If another owner member tries to review the message, a 400 error is thrown that the message has already been reviewed.

VALIDATION:

"*Between the acceptance criteria and this use case, I think that you have captured it well, I think its good that the message is sent off to be approved so then each message is double checked for suitability in the chat*"
- Jess King

Interface Design

Name & Description	HTTP Method	Data Types	Exception
message/reportmessage/v1	PUT	**Body Parameters** (messageId) **Return type if no errors** {}	**400 Error** when: ● messageId does not refer to a valid message within a channel/DM that the authorised user has joined
message/reviewreportedmessage/v1	POST	**Body Parameters** (messageId, ogMessageStr?, reactId) **Return type if no errors** {}	**400 Error** when: ● messageId does not refer to a valid message within a channel/DM that the authorised user has joined ● The reported message has already been reviewed by another owner member ● reviewId is not a valid reviewID - 1 = accepted, message is edited to it's original form ● 2 = denied, message is deleted **403 Error** when:

			● the authorised user does not have owner permissions in the channel/DM that the message was retrieved from

correct Email + Password + registered

Not logged in

message deleted

incorrect info

owner does not approve of message

Logged in

message reported

channel info

owner approves of message

owner member is notified of flagged message

channel created

channelId, message

mesageId

message sent

mesageId

message is displayed

owner of channel deems if message is appropiate