Assignment M5

Jill Ellington
jmatthews87@gatech.edu

Abstract—During the COVID-19 pandemic, Microsoft Teams usage as a collaboration platform in the workplace has skyrocketed from 20 million users in 2019 to 145 million users in 2021 (Curry, 2021). It is now an integral piece in collaboration teams - remote or otherwise - but can it be improved? This study focuses on how Microsoft Teams generates and employs notifications to alert the user of new information within its chat feature, teams/channel feature, and calendar/meeting feature.

1 QUALITATIVE EVALUATION

For this qualitative evaluation, I executed a survey where users evaluated a textual prototype describing a scrolling banner for various Microsoft Teams notifications. The users were asked to read the textual prototype (*Appendix 5.1: Textual prototype for scrolling notifications banner*) and answer 7 follow-up questions. The survey questions can be seen in *Appendix 5.2: Survey questions*.

The survey was offered on the Georgia Tech Peer Survey tool, and received a total of 13 responses. Considering there was a bit to read in the textual prototype, I was satisfied with this number of results. If I did change anything for a future qualitative evaluation, I would have gathered feedback from a wider audience than just Georgia Tech students.

1.1 Raw results

The questions in this survey were a mix of types, and each were summarized differently (see Table 1 below). The comprehension questions tested how well the textual prototype described the intended functionality (ie, how well the users understood how it would function). The summarized results capture what percentage of participants correctly summarized the functionality and the percentage of those who did not. The quantitative question asked how satisfied the users were with the interface as described (1 is the least satisfied, 5 is the most). And finally, the qualitative questions were open ended and were about how the users liked and disliked the interface, and what suggestions or comments they had about it. For these questions, the results are summarized into

themes and ideas. If the theme or idea was mentioned more than once, I included the number of times it came up next to it.

Table 1—Summary of survey results. Raw results can be seen in *Appendix 5.3: Raw survey results*.

Question	Type	Result
1. Notification banner usage	Comprehension	62% correct
		38% incorrect
2. No more notifications	Comprehension	54% correct
		46% incorrect
3. Satisfaction with interface	Quantitative	2.77/5
4. What do the users like	Qualitative	Fixed location for all notifications (5)
		Nothing (2)
5. What do the users dislike	Qualitative	Number of notifications likely to occur (4)
		Distracting/cluttered (4)
		Speed of reading may vary across users (2)
6. User suggestions	Qualitative	Grouping notifications based on user or type
		Scroll once, then user action to see again
		Collapsible
		Use eye tracking for speed of reading
		None (6)
7. Other comments	Qualitative	Option to turn off for privacy reasons
		None (9)

1.2 Feedback and takeaways

The first takeaway I had from this survey is that I might not have adequately described the functionality through the textual prototype based on the responses of the two comprehension questions. On the other hand, several of the interface suggestions I received were about the type of prototype, rather than the interface in the prototype, so I suspect that users were getting burned out reading textual interfaces for surveys, which is understandable. From that feedback alone, I

might look at skipping a textual prototype in the future to focus on more visual or interactive prototypes.

Another takeaway is that in general, users would not like to see this prototype implemented. The question about user dislikes resulted in many more responses than the question about user likes. The best thing that can be said about this interface is that it provides a central location for all notifications in Microsoft Teams. After that, everything described is negative. The survey participants believe the notifications provided in a scrolling banner will be too distracting, cluttered, numerous, and ultimately, not useful. I can't say that I was surprised by this feedback. One of the reasons I chose this prototype for survey evaluation is that I knew there were significant improvements possible, and was interested in hearing the opinions of others.

What I was most surprised by in the survey results were that I received a number of useful suggestions for how to improve this. While I was hoping for this, I didn't expect the number or quantity of good ideas I received.

1.3 Prototype changes

The feedback from this evaluation provides a number of useful changes.

One of the negative aspects to this design is that each user reads at a different pace, so the scrolling text might be either too fast or too slow. This could be mitigated by tracking users' eye movements to adjust the speed of the text.

Another change that several users suggested is a way to collapse or hide the notification bar. This could be accomplished with either a chevron to allow collapsing of the scroll bar, or adding a setting that disables it permanently (the use case for this would be privacy related - if a user never wanted notifications to scroll to protect theirs or others privacy).

2 PREDICTIVE EVALUATION

I will be performing a cognitive walkthrough of my Wizard of Oz prototype for a voice interface, seen in *Appendix 5.4: Wizard of oz prototype*.

I will use **bold text** for voice commands that I issue as the user, and *italicized text* for responses from the voice interface. Any other voice commands or responses that I include as suggested improvements will not be marked that way.

2.1 Cognitive walkthrough

As a user of Microsoft Teams who is a novice to this voice interface, I probably wouldn't know that there is a way to alert the application to listen for a command by saying "Listen up, Teams!" However, there will be a dedicated button in the interface with a microphone, and I use that signifier to assume that is how I issue a command. I click it, and the rest of the interface goes slightly grey, while the button changes to green to indicate active listening.

I say **"What notifications do I have?"** That is close enough to the available command ("What are my notifications?") that the interface responds correctly with the number and type of notifications I have. It says *"You have 3 new chat messages and 1 unread Team channel message".*

Not knowing exactly how to proceed, I say **"Describe those notifications in more detail."** That is not like an existing command, so the interface responds with *"Unknown command. Say "Help" for a list of commands to use."* I realize maybe there should be a way for the user to find possible commands without needing to make a mistake first. Perhaps the first couple of times the voice interface is activated, it could make suggestions for how to proceed. For instance, it could append, "To listen to your chat messages, say 'What are my Chat notifications?'" to the end of appropriate messages.

I say **"Help"** to understand what I can say. It responds with *"If you would like to hear your chat messages, say 'What are my Chat notifications?' If you would like to hear your Team channel messages, say 'What are my Teams notifications?'"*

I realize that the fact that the product is called Microsoft Teams and there is an area of the app called Teams is confusing. Bridging both the gaps of execution and evaluation may be difficult here for the users. As a suggested improvement, the interface should use the voice command "What are my channel notifications?" as an alternative.

I say **"What are my Chat notifications?"** The interface responds with *"You have 3 new chat messages. Joe Smith says 'How about we bring lunch in for the noon meeting?' Barbara Cane says 'Thanks for sending along that important document!' In your group chat titled 'Team Spyro', Andrea Thompson says 'Hey Team! When is the best time to schedule our next refinement session?'"*

This is a lot of information being read back at me at one time. As an alternative, perhaps the interface could read the first message before asking what to read next. Or list the 3 senders then wait for the user to ask for a particular chat message.

I also wonder what the rest of the app is doing while these voice interactions are occurring. Is it still greyed out? Or can the user navigate around while the voice interface is responding to questions? I wonder if as the interface is answering a voice question, it can also provide the ability for the user to jump to the specific message being read so they can respond if they wish.

Another consideration is: Should the user be able to issue voice-to-text responses via the voice interface? I only designed this to be a notification interface, but it could easily incorporate ways to respond to chats and messages as well.

At this point I expect a user might be annoyed with the verbosity of the interface and go back to using the app sans voice commands.

3 EVALUATION SUMMARY

3.1 Understanding the user

There are several ways that I would want to understand the user better.

I would be interested in knowing how many users of Microsoft Teams have a vision impairment and/or have difficulty using the application in the traditional way. Do they have a need for a voice interface, or would a scrolling notifications banner help or hurt a screen reader?

I would also be interested in hearing more about the usage of Microsoft Teams by users who sit in an office with others versus those who primarily telework. Would a voice interface be less annoying to those who are teleworking?

3.2 Design alternatives and brainstorming

Both evaluations brought to mind additional design alternatives, specifically in the capacity of slight changes to the existing prototypes.

3.2.1 Scrolling notifications banner

The scrolling notifications banner received mostly negative feedback. I would create a few alternative prototypes showing the scrolling notifications in different

ways, such as different groupings and various configurable options like collapsibility/hideability. This design change would remain at the current level of fidelity to result in a faster feedback loop to hone in on the best improvement options.

Another design alternative I would be interested in adding is an eye tracking capability so the scrolling text speed can match the users' reading speed. I would be tempted to raise the level of fidelity of this design to create a more realistic interface for better evaluation, but given the negative feedback for this design in general, that would likely be a waste of time and resources.

3.2.2 Voice interface

The voice interface would obviously perform better with a more robust conversational interface, specifically by allowing for better initial affordances and signifiers so that the user can bridge the gaps of execution and evaluation more readily when they are new to the interface.

The design should also incorporate more concrete design for what would happen in the rest of the application while the voice interface was being used.

I would keep the fidelity the same for this design change, again to enable quick feedback for future, higher fidelity, prototypes.

3.3 Next evaluation plan

I don't think either prototypes would be ready for empirical evaluation, and would pursue more feedback based on qualitative methods. My reason for choosing qualitative over predictive is that I prefer actual users' feedback to my own, to avoid as much personal bias as possible.

4 REFERENCES

1. Curry, D. (2021, May 5). *Microsoft Teams Revenue and Usage Statistics (2021).* BusinessofApps. https://www.businessofapps.com/data/microsoft-teams-statistics/

5 APPENDICES

Appendix 5.1: Textual prototype for scrolling notifications banner

This prototype describes a scrolling banner across the top of the Microsoft Teams app that displays current notifications to the user.
The banner would be in a fixed location at the top of the app. The space would always be there so there would be no doubt whether there were any notifications or not. If it is blank, then there are no notifications.
When there are notifications, they would appear in the banner. The notifications would scroll across the banner so as to provide the maximum amount of information. The entirety of the text would not necessarily fit in the banner at any given point, hence the scrolling. For instance, if there were multiple chat notifications, the banner would scroll the text "Chat from Joe Smith. Chat from Elizabeth Burl. Chat from Mike Pill." until the user reads the chat messages, at which point they'd drop out of the notifications banner. Similar text would appear for unread channel message notifications and calendar notifications. An example that employs notifications from all three areas of the app might be "Chat from Joe Smith. Unread message in Users Group team > Announcements channel. New meeting called Status Updates scheduled for tomorrow at 9am."

Appendix 5.2: Survey questions

1. For the next few questions, imagine you are using the Microsoft Teams interface with the feature described in the textual prototype.
   Please describe how you would consume application notifications. (Short answer)
2. How would you know if you have consumed all of the application notifications? (Short answer)
3. Do you agree with this statement? I would be satisfied with the scrolling notification banner as described in the prototype. (Agreement scale of 5-1)
4. What do you like about this interface3? (Short answer)
5. What do you dislike about this interface? (Short answer)
6. Do you have any suggestions for how to improve this interface? (Short answer)
7. Do you have any other comments to provide regarding this interface? (Short answer)

Appendix 5.3: Raw survey results

[{"id":"1626023997628","text":"This textual prototype provides an alternative method of generating user alerts in the Microsoft Teams communication tool. Please read the following text and answer the questions below. This prototype describes a scrolling banner across the top of the Microsoft Teams app that displays current notifications to the user. The banner would be in a fixed location at the top of the app. The space would always be there so there would be no doubt whether there were any notifications or not. If it is blank, then there are no notifications. When there are notifications, they would appear in the banner. The notifications would scroll across the banner so as to provide the maximum amount of information. The entirety of the text would not necessarily fit in the banner at any given point, hence the scrolling. For instance, if there were multiple chat notifications, the banner would scroll the text “Chat from Joe Smith. Chat from Elizabeth Burl. Chat from Mike Pill.” until the user reads the corresponding chat messages, at which point they’d drop out of the notifications banner. Similar text would appear for unread channel message notifications and calendar notifications. An example that employs notifications from all three areas of the app might be “Chat from Joe Smith. Unread message in Users Group team > Announcements channel. New meeting called Status Updates scheduled for tomorrow at 9am.”","answers":["I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype.","I have read the textual prototype."]},{"id":"1626031021679","text":"For the next few questions, imagine you are using the Microsoft Teams interface with the feature described in the textual prototype. Please describe how you would consume application notifications.","answers":["Reading them as they pop up ","Whenever I would want to check my notifications, I would look up to the scroll bar","I am actually not very clear how this is different from the current notification banner","I would use the existing notifications feature in teams as I like the current design.","I mean I guess I would look at the top of the application as I was using it to look or I would go to my unread channel.","When I get a notification, if I'm not too busy with something else, I will click it to open whatever I'm being notified of.","Interesting concept but I prefer the current

notification method. Each individual meeting chat or personal chat will keep its notification. ","not sure what you mean by consume, acknowledge? according to your description you just watch it scroll.","I would look at the banner to see if there are any messages that I missed","Easily since it seems the banner will give me the resource to do this easily","I think this would be alright if you could click on it for the full notifications. I would use it as a reminder, not really reading the notifications themselves","I would check the banner regularly to see if I had any outstanding notifications.","read them across the top"]},{"id":"1626031040675","text":"How would you know if you have consumed all of the application notifications?","answers":["The banner no longer has any text ","Potentially some sort of visual break between the notification loops, it could be as simple as a pipe | or it could be more fun like an emoji","when they are all disappeared ","It is not clear from the description.","My unread channel gets cleared after I read it","The notification area is blank","I would not as I would be concerned that I missed a scrolling notification. Especially with multiple activity items while I was inactive","the scrolling repeats?","No notifications are displayed","Not sure, probably implement a system where they are marked off","I'm not sure, that there were none left?","I would know I had seen all notifications if the banner was empty.","it drops out after it's been read"]},{"id":"1626031048697","text":"Do you agree with this statement? I would be satisfied with the scrolling notification banner as described in the prototype.","answers":["2","3","3","1","2","3","2","4","2","4","3","4","3"]},{"id":"1626031055574","text":"What do you like about this interface?","answers":["Consistent location is useful. ","I like that it wouldn't be as initially interruptive as the big banner notifications","I am actually not very clear how this is different from the current notification banner","N/A","Tries to alert all of my missed notifications in one spot","I like the idea of having notifications easy to access","Its an intersting concept","scrolling text","The banner displaying the notifications","Scrolling and fixed location ","I think it would look pretty cool","I like that there's one place to view all my notifications.","seems lively"]},{"id":"1626031202812","text":"What do you dislike about this interface?","answers":["Group chats are likely to cause a huge number of notifications. Since the interface is a fixed size, there might be a notification that comes in that would take several seconds to digest because of 500 group messages. ","If it keeps scrolling even after I have read the notifications, that may become distracting. If I am in the middle of something and can't directly check the chats that it is notifying me about, the scrolling would probably be bothersome","assuming that this means everything will be

shown rather than one by one as it is currently, i find it too cluttered ","I don't like the scrolling notification as it is not clear if it will read at my pace and also may distract me. It is also not clear how to dismiss a notification.","I already get notifications that pop up on my screen while I am using the app.","I dislike the idea of the notification area always being there.","It would require more attention from me to see all of the notifications. ","textual prototype","The scrolling notifications. Wish there was a way to see all notifications at once.","no clear notification button","depending on reading speed it might get annoying","I think it could be distracting, especially if I have a lot of notifications.","could be annoying"]},{"id":"1626031214228","text":"Do you have any suggestions for how to improve this interface?","answers":["Group the notifications based on who is sending it. ","I think potentially having the notifications scroll through once could be beneficial and then if the user hovers their mouse over the scroll bar, the notifications could re-scroll so they could read them, that would help reduce unnecessary scrolling if the user has already seen the notifications, but just can't act on them right away","Nope, I quite like the current interface","N/A","N/A","It should have a button to collapse the area until a new notification arrives. There should also be a preference setting where a user can turn off this feature if they want","NA","use paper prototype","Provide a way to see all notifications. Maybe give the ability to click banner to see all","add a clear notification button","eye tracking for reading speed","N/A","NA"]},{"id":"1626031221673","text":"Do you have any other comments to provide regarding this interface?","answers":["Scrolling text is very distracting -- maybe there is another way to create a visual representation of a new message that can be revealed rather than shown by default. ","Fun idea to potentially improve work!","Nope ","I find the current teams notification interface good.","N/A","no","NA","n/a","No","Option to not show message for privacy reasons","nope!","N/A","NA"]}]

Appendix 5.4: Wizard of oz prototype

In order to use the voice command feature, the interface needs to know a voice command is about to be issued. There is a button with a microphone icon in the interface that the user presses prior to issuing a voice command. The user could also say “Listen up, Teams!” to signal the interface to listen for a command.

Once the interface is listening, the user can make the following requests:

- "What are my notifications?"
  The interface would respond with the number and types of notifications. In this context, types of notification would be chat, teams, or calendar notifications. For more detail, the user could ask follow-up questions below.
- "What are my Chat notifications?"
  The interface would respond with the number of chat notifications, and then start listing who they were from. For instance, if Joe Smith has sent a chat, the interface would say "1 chat message from Joe Smith".
- "What are my Teams notifications?"
  The interface would respond with the number of channel notifications, and list which channels they are in. For instance, if the Team Announcements channel in the ABC team had unread messages, the interface would say "Unread messages in the Team Announcement channel in the ABC team."
- "What are my Calendar notifications?"
  The interface would respond with the number of calendar notifications, and a list of the new or updated meetings that correspond to those notifications.
- "Help"
  The interface would respond with a list of voice commands it can respond to.
- Unknown voice command
  In the event that the user issues a command that the interface doesn't understand, it would indicate that with "Unknown command. Say "Help" for a list of commands to use."