Creating Knowhow

Vinita Atre, Olivia Garahan, Rachael Marr & Di Xiao

Our team is addressing collaborative design with older users. 20-22% of the U.S. population will be over 65 by the year 2030, and their capabilities differ greatly from those of younger users. Age-related decline of both cognitive and physical ability has many implications for older generations' technology use.[1] Variations in cognitive ability are often overlooked in user interface design, but are especially important to consider when it comes to teaching a user with diminished cognitive capability a new system. As people age, working memory declines. Seniors are often not digital natives, so navigating an interface that would be second nature to a younger user often results in an older user having to rely on their working memory to remember what certain icons and other visual shortcuts mean.[2] In T.A. Hart et al.'s study, researchers found that information overload was common, especially in novice senior users. This overload is often a combination of too much information needing to be stored in working memory and a lack of ability to easily access basic information about completing tasks that a younger user would simply search for.[3] Older people also tend to be less able to recover from user error, and are therefore overall more hesitant in interacting with an interface, spending a great amount of their interaction time simply considering where to go.[4]

In creating an interface for older users, collaborative design is essential to success. In D. Hawthorn's article "Interface design and engagement with older people", researchers attempted to develop a successful interactive tutorial for older people. Throughout their design process, the researchers found that inconsistencies between designers' assumptions about the way users will interact with a system and the way that older users actually interact with a system were greater than expected, further driving in the point that stakeholder participation in this context is vitally important.[5]

The need for collaboration in design reaches as far as the design process itself. In the above-referenced Hawthorn paper, researchers identified that older users have some difficulty interacting with low-fidelity prototypes, as many of the small design considerations that are often made later in the process are not included in a low-fidelity prototype.[6] This was reflected in our second session with the user, who identified several issues with the low-fidelity prototype that she found confusing, but that were only extant because of the fact that she was interacting with a prototype and not a real system.

The system we created is a digital assistant in the form of a chatbot that users can query for help. When delivering assistance, the system highlights the relevant parts of the page to demonstrate to the user how to perform their task. This provides an easy-to-follow explanation of how to complete a task that is demonstrative, visual, and textual. The user does not have to leave the page they are working on in order to understand how to complete a task, and the solution is presented to them in a simple and understandable fashion.

Tasks

For this tutorial system, the three primary tasks we addressed with our high-fidelity prototype were:

1. Figure out the system capabilities;

   On opening a web application, user can get an overview of the system and understand its capabilities if they choose to watch a quick walk-through. This task can be crucial in determining if the users want to use the system.

2. Ask how to create a new Google Document;

The user would need to be able to ask and receive specific answers to her questions through interaction with the chatbot. A sequence of highlights will also be shown on the page as a tutorial.

3. Learn how to perform document operations;

Users can learn to perform detailed operations on one page. We focused on operations such as: print, save, and download. The participant expressed she would like to know how to work with a document, because autosave is a concept that the user may not understand, especially in the context of Google Drive.

Interface Revisions

In the second session with the user, we tested two prototypes. One was a left-sidebar chat design (Design One), and one was a pop-out chat design (Design Two). Design One tested well with the user, garnering an overall approval rating of 7/10, while Design Two had an overall rating of 4/10. These designs were both used as a jumping-off point to collaborate with the user to create a third interface, which was the design that carried our team forward into our final high-fidelity prototype. Designs One and Two can be seen in Appendix A. Design Three sketches can be seen in Appendix B.

The Design One prototype was able to, via chat, walk the user through one task. It also included the option for a walkthrough once the user had completed a task. During the Design One session, there was a critical incident (rated as a three) in which the user could not find the sidebar to begin with. This was despite the fact that the proctor showed the participant what the real Google Drive looked like prior to performing the test, and did a quick user walkthrough of the system, most notably the sidebar functionality itself. Despite this, the user struggled to find the sidebar when it was their turn. However, once the participant became used to using the sidebar they enjoyed it, "I was just confused--before--but I like this now... I can collapse it whenever I want..." The second critical incident (rated as a two) was how fast the mock version ran. In the prototype, the GIF which simulates a chat was a little too fast for the user to follow. Although this did not directly affect our final prototype, it was important to take note that the user was not thrilled with the speed of the system, and felt that they did not have full control over how the system runs.

The user appreciated a lot of the aspects of Design Two, but gave the system and overall rating of 4/10. This was due to three critical issues that were found. Issue one (rated as a three on the severity scale) was that the user did not understand the dropdown options that were given, making it very difficult for the user to begin to use the system. She asserted, "I would probably just select open, since I don't know what the options mean..." The second issue (rated a one) was the rollover of the "function" option. The user did not understand what this meant in the context of Google Drive, and she found the rollover "irritating." Once the options were explained more in depth to the participant she conceded that the function was useful, but stated that she was more concerned with performing actual tasks, and that she could understand whatever the rollover would have to teach her just by using Google Drive. The last critical incident, rated as a 3, was the way everything was placed on the screen. The participant felt that the chat window being on the right and the instructions on the left were confusing, stating that, "it's really jarring that I am going back and forth..." This prototype did have its good points: namely, the participant liked that the explanation never left the screen, the overlay on the system, and the size of the typeface. However, it was too difficult for the participant to feel like they could keep track of everything that was happening.

The team decided to go with a third design option that we worked with the participant to design (see sketch at right). The participant wanted to keep the sidebar idea, but was very adamant that the sidebar be a color that gave the user the impression that it should be clicked, and also have some sort of wording to pop out

the sidebar. The participant also introduced the idea of a button on the screen that would connect to the sidebar (and would pop it out when needed). The participant was very concerned with the perception of the sidebar being clickable and felt that with the addition of the button, people would know how to access the tool better. Design Three also included a voice option, which the participant felt might be useful for some users, herself included, although she stated that she would still like to see the words in the chat. She also liked the idea of an overlay on the screen itself that would highlight a section, which was seen in Design Two.. The participant appreciated the separation of the background to the action that the chat was telling her to perform. In addition, the participant talked about how she wanted the color to be bolder, and, if possible, to match the color of the action that needs to be performed in the chat itself. For example, on Google Drive, the button for anything new is blue, the participant wondered if we could incorporate that color into the directions of the chat so that it connected more.

Prototype Overview

The KnowHow prototype is based on providing users with clear and concise answers to their questions. As older users usually prefer to talk to their friends and family to get brief and instructional answers, we decided to incorporate the same. KnowHow is aimed to integrated in the web browser. Visually it is left aligned and occupies same height as the application. It contains chatbot interface that takes user input and guides user in performing tasks. The chatbot is not hard coded but is based on an intelligent agent. Our intelligent agent was named Alex as shown in fig 1. The intelligent agent is hybrid of rule based and machine learning. Due to the scope of creating a prototype, we did not train the agent using large dataset and take full advantage of the machine learning aspect. We defined rules in the intelligent agent as shown in fig 2. Knowhow uses Natural Language Processing (NLP) in its chatbot interface so that they can ask questions in a natural, conversational way. What this means is that user can ask questions like " how do I create a new doc", or "how do I get this doc on my computer?" The first question refers to the task of creating a new Google Document. The second task refers to the task of downloading a file. For tasks involving user interaction, along with the chat response, a walkthrough is displayed using an array of screens with visual cues for easily learning the interaction. So if the user wants to know how to download a document. We provide them a textual instruction in the chatbot window. In addition, we also highlight the buttons to be clicked on the application screens in the appropriate order. These highlights are shown using the technique of greying out the entire screen and only showing the button which is the current call-to-action on the screen. In case of downloading a document, the application screen will highlight "file" option in the menu. It will then show the next state after clicking file option which is the drop down menu. In the drop down menu, the "download as" option will be visually highlighted. These highlights tell user the buttons that they need to click. As procedural learning is relatively easy with older adults, we show them a step-by-step visual procedure of how to get a particular task done. We also provided users with a video overview when they opened a new web application. The video is not auto played and the user can play or skip them as per their need. The chat interface is displayed along with the web application screen to allow for easy access and referencing. We also placed a sidebar to clearly separate the chat interface. The chat interface was placed to the left of the application occupying the total height of the web browser as many web applications have chat interfaces in-built and these are placed to the right and we wanted to make our chat interface distinguishable.

Screenshots and diagrams of the creation process of the prototype are available in Appendix C.

Description of Technology Used

KKnowHow prototype comprised of two sections. The first was the chatbot interface on the left and the second was the actual application space on the right. The

chatbot interface was developed using Javascript client. An open source javascript client was used and customized to meet the needs of the prototype. The application section is developed in html/css and javascript. The entire system is hosted on a web server. We used the http protocol over localhost for our requirements.

On the backend, the chatbot interface communicates with the Api.ai service. We created an intelligent agent called with the Api.ai service. We named our agent Alex. The api.ai provided us a access token for our intelligent agent.We gave rules to the AI agent and determined the possible text responses. Rules are created in api.ai using intents. Every intent has a corresponding text response, image response, JSON response and event associated with it. We also showed these image responses as an array of image responses to make it easier for the user to understand the interaction on all screens. The images were hosted on the amazon S3 server to provide easy access and modularity. The JSON response was used to send video url in chat response. The video url could correspond to any video on youtube. We tried to keep the chatbot response separate from our front end view. Defining rules and responses was made modular and only required one to access the intelligent agent on Api.ai service.

The chat interface also used a Text-To-Speech(tts) module to read out the responses from the AI agent. The tts had a female voice. The tts module was integrated in the javascript client for the Api.ai service. So the text responses sent by the intelligent agent are parsed from the JSON response message. These text messages are sent to the tts module which reads them out to the user. Through heuristic evaluation, we found the need of giving users the option to mute and unmute the spoken response. A simple switch button was provided to the users in the chatbot interface to mute and unmute the spoken response.

The intelligent agent was helpful in creating a robust and user friendly help system by supporting machine learning and natural language processing. The agent also supported a variety of responses including text, images, events and custom JSON. The JSON response provides easy integration on the javascript client end. It also provides great deal of flexibility in send response data.

However, we wished that the tool had an analytics component. It would help us to understand user behavior and interactions. This would also help us in defining better rules for our intelligent agent.

What Wasn't Implemented

However, we did not implement the Speech recognition module to ask questions to the chatbot. We did not anticipate the value that speech recognition had for the user.

We also had limited set of functional questions that our AI agent understood. Since the model was aimed to be used on a prototype, we did not focus on being comprehensive in creating rules for AI agent. The process is however simple and can be easily extended to make the chatbot more knowledgeable

We only designed the Agent for Google Drive and Docs. We focused on understanding the chatbot interactions and user acceptance. Further steps include studying chatbot help for different kinds of web applications.

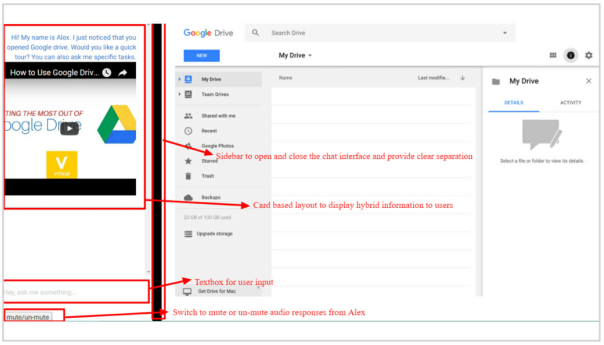
Storyboard

1 Working as lecturer in Arts, Paula has to use web applications like Canvas and Google Suite

She calls her daughter for help

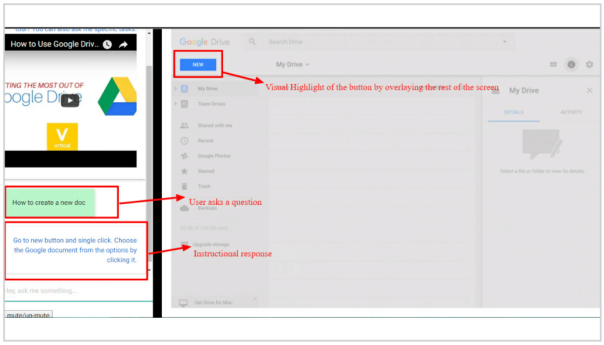
2 Paula is working with Google drive and docs

She decides to use the Knowhow tool to get more agency



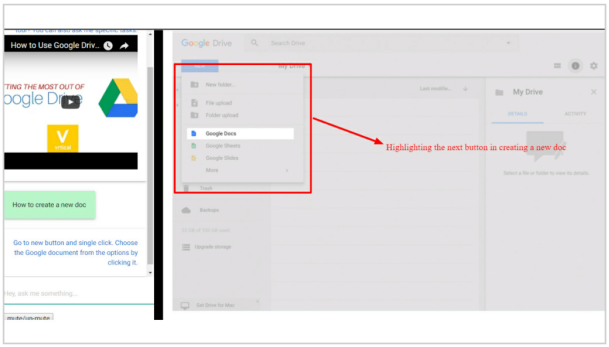
3 Paula opens Google drive and wants to get an overview

Knowhow opens a chat interface gives the user a video overview



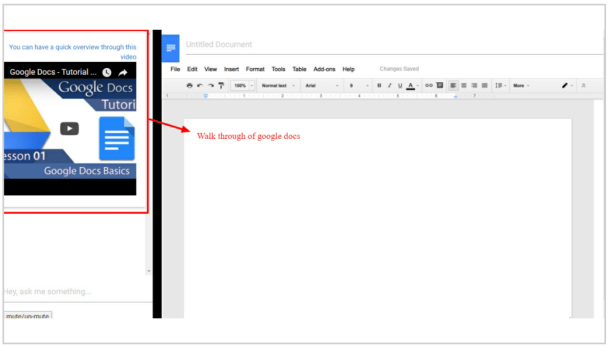
4 Paula does not know how to proceed so she asks Alex

Alex gives her instructions and provides visual highlights on the application.



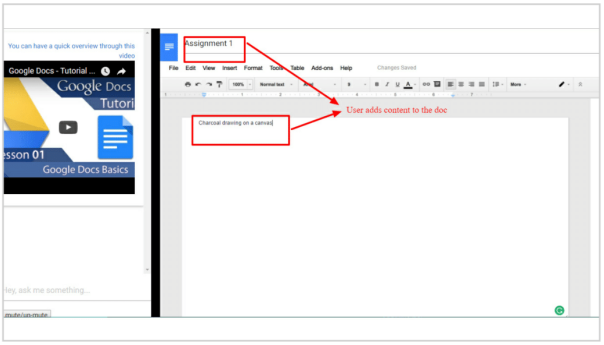
5 Paula follows the visual highlight clues

Alex gives her instructions and provides visual highlights on the application.



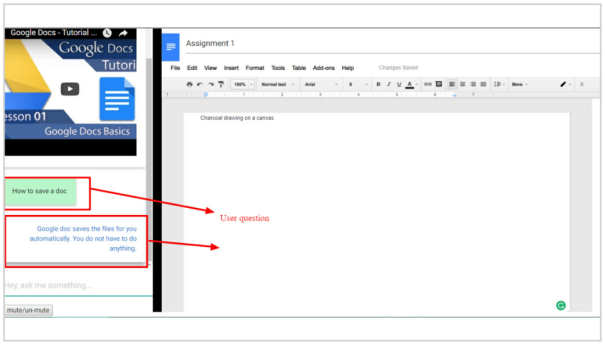
6 Paula creates a google doc

Alex gives her overview of the google docs. She chooses to skip it.



7 Paula adds content to her document

Alex gives her overview of the google docs. She chooses to skip it.



8 Paula wants to save the document. She asks Alex

Alex explains about saving feature on google docs

Evaluation

In the protocol we began by revisiting what we learned with the user in the last session. In that last session we learned:

The participant wanted the system to be based on her actions

The participant preferred the sidebar design

The user wanted to learn more about Google Drive, and specifically Google Docs.

The user agreed with what we asked, and added that she also did not like the speed of the prior system. She found that very jarring and hard to understand. In addition she reminded us that she did not like the type-size of the prior versions, because they were very hard to see.

Next, we had the user complete five tasks. Those tasks are:

0) Explore and understand Google Drive

1) Create a Google Document

2) Edit a Google Document

3) Learn how saving works with a Google Document

4) Learn to download/print a Google Document

These tasks took the participant 8 minutes and 46 seconds to complete. Afterward, we had an open-ended discussion about how the participant felt about the system. Overall the user responded well to the prototype. She gave the prototype a 4 out of 5, which was a lot more positive than our lo-fi versions. The user had a little bit of trouble seeing where she needed to ask the system something, and hesitated a little bit when there was no "send" button for her to click after she had typed her question.The user had trouble when creating a new document as the submenu was confusing to the participant, but she was able to retype her question and see the animation of what to do again. The other task that was confusing was printing the document. The proctor had to reframe the question so that she would not perform the action she already knew how to do. There were no other issues with tasks and the user was able to complete those quickly.

In the open-ended discussion the user was adamant that the system would fit her needs and even asked if it was available for the entire google suite. She had a couple of improvements she would like to see. Firstly, she would have loved to have been able to speak to the system as well, she enjoyed the voice portion (her idea from the last session), but she wished she could interact with it via her voice as well. She also felt that the "hey ask me something" portion should be more visible. Stating that, "maybe it could light up or something?" That was an issue, because she was not able to find that section of the design initially. In the positives, she felt that she could use this system, and was excited that we took her advice for the final design. She thought it was easy to use. She did give the system a rating of four out of five only because she really had wanted to use her

own voice when interacting with the system.

Session notes are available in Appendix D,

Design Process Reflection

There were a lot of things that we were surprised by, and excited about when the user gave us their opinion. One of the best design decisions, the addition of the voice, was incredible insight, and not something any of us had thought about before. We also added a name to the system once the user talked about how she always asks her daughter for help. We named the system Alex. However, looking at the entire design, the look and feel of it did not change under the user a lot.

Participatory design does have some limitations. Specifically, you are only working with one user, which means that your product might not be palatable to others, even though they are in the same user group. Also, when coming up with something like a chatbot as a solution, the participant could not understand how a chatbot worked. So we could not rely only on the ideation of the participant as they did not know the possibilities of the technologies as well as we do (just by the nature of our work.) This is where the solution has to come from the team's experience and be applied to the user group.

As for our evaluation, it might have made more sense to do paper prototypes for the low fidelity versions of the tool. However, the user was not familiar with google drive so it would have been hard to test on paper to gauge the participants understanding of the system. For newer technologies, it might be worth it either test a more robust "dumbed down" version of the tool, or run a paper chatbot on something that the user already understands. As for approach, contextual design might have been better. That way we could have gotten detailed information from several participants, and then giving the system more buy-in from a wider array of actual users. Contextual design is especially useful when looking at the diversity within the age group we have chosen, it gives us the opportunity to work with more than one user, and begin to take into account the differences within this population.

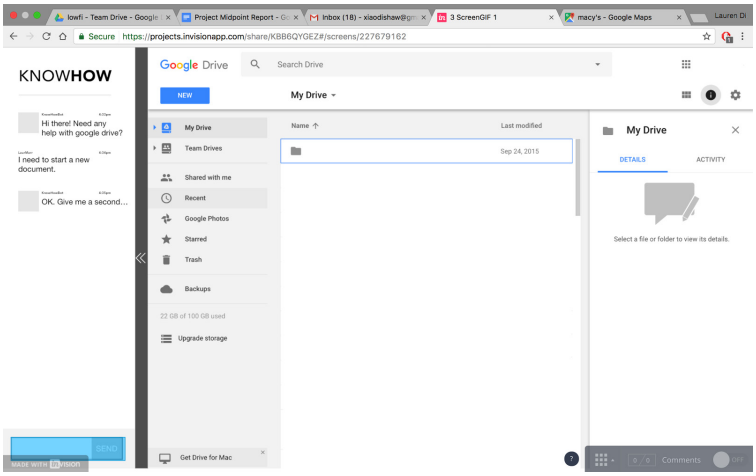
The most valuable methods we used were the initial user interview and the prototype testing. Those two portions gave us a lot of insight into the environment that the user would be using the product in. In general, the least useful was testing the low fidelity prototype as the technology made this difficult.

Works Cited

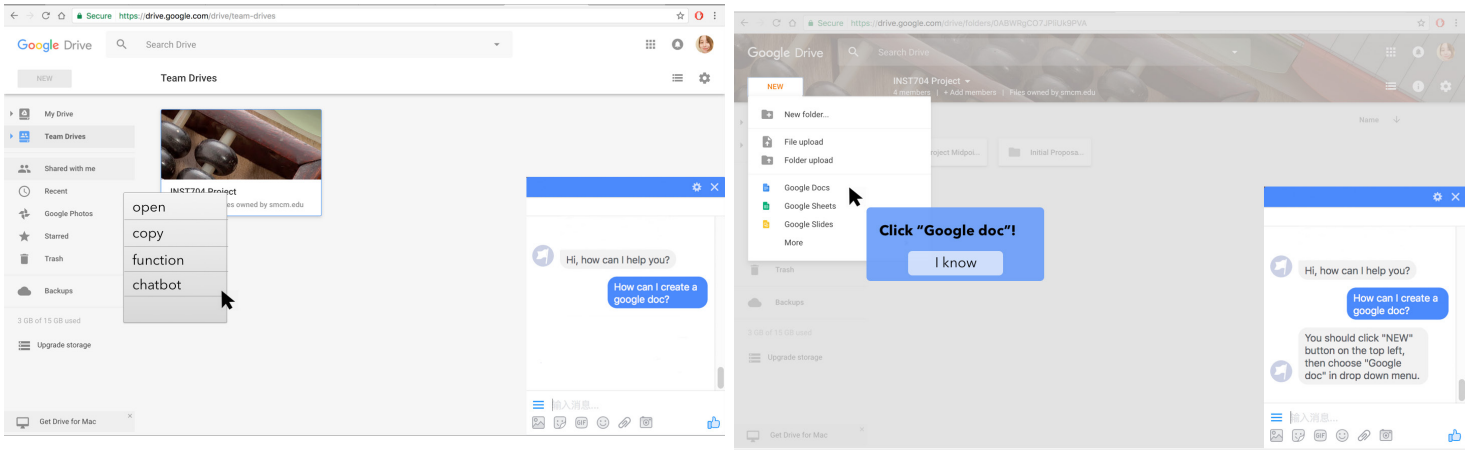
1. Smith, Aaron. "Older Adults and Technology Use." Pew Research Center: Internet, Science & Tech. Pew Research Center, 03 Apr. 2014. Web. 14 Mar. 2017.
2. Hart, T. A., Chaparro, B. S., & Halcomb, C. G. (2008). Evaluating websites for older adults: adherence to 'senior-friendly' guidelines and end-user performance. Behaviour & Information Technology, 27(3), 191-199.
3. Charness, Neil, and Walter R. Boot. "Aging and Information Technology Use." Current Directions in Psychological Science 18.5 (2009): 253-58. Web.
4. Mayer, Jeff. "Senior's Guide to Computers - A Beginner's Tutorial for the Microsoft Windows PC, Hardware, Software, Security, Accessibility, Internet, Backup and Email." Senior's Guide to Computers - A Beginner's Tutorial for the Microsoft Windows PC, Hardware, Software, Security, Accessibility, Internet, Backup and Email. N.p., 12 Apr. 2007. Web. 14 Mar. 2017.
5. Gregor, Peter, Alan F. Newell, and Mary Zajicek. "Designing for Dynamic Diversity."Proceedings of the Fifth International ACM Conference on Assistive Technologies - Assets '02 (2002): n. pag. Web.
6. Schuler, Douglas, and Aki Namioka. Participatory Design: Principles and Practices. Boca Raton: CRC, 2009. Print.
7. Wen Huei, C., Yu-Ting, L., & Kuang-Hsia, L. (2013). User requirements of social media for the elderly: a case study in Taiwan. Behaviour & Information Technology, 32(9), 920-937.

Appendix A

Screenshots of Designs One and Two



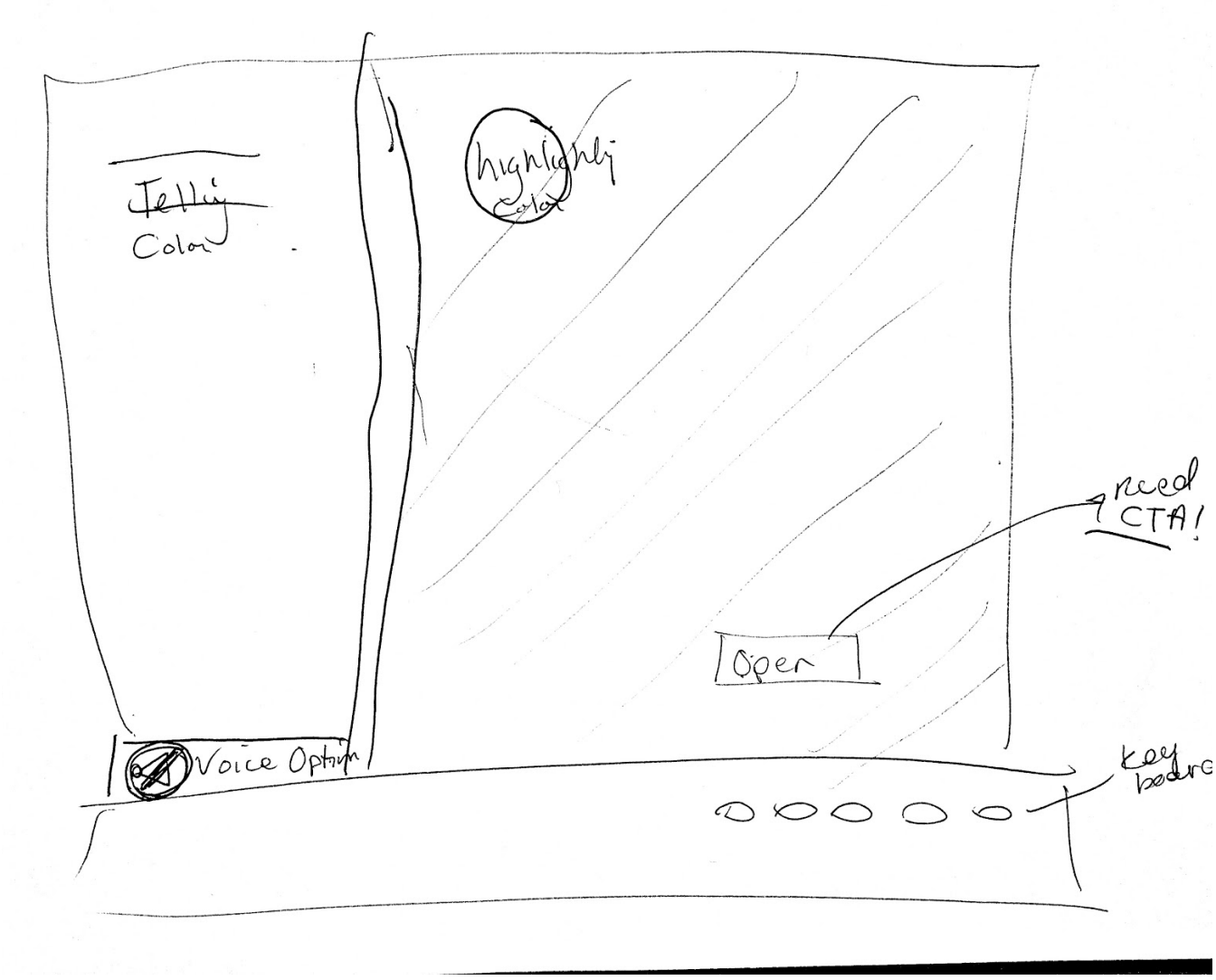
Two typical pages of Design One; (left) a pop out window with clear answers to users' questions; (right) a walkthrough for users to find more information



Two typical pages of Design Two; (left) the drop down menu designed for each button; (right) the animation is presented and unrelated content turns into grey

Appendix B

Sketch of Design Three



Two typical pages of Design Two; (left) the drop down menu designed for each button; (right) the animation is presented and unrelated content turns into grey

Appendix C

Process screenshots

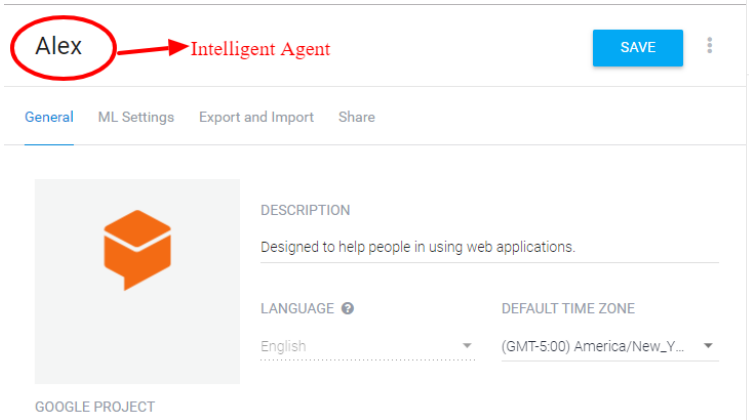


Figure 1: Creation of AI Agent on API.AI service

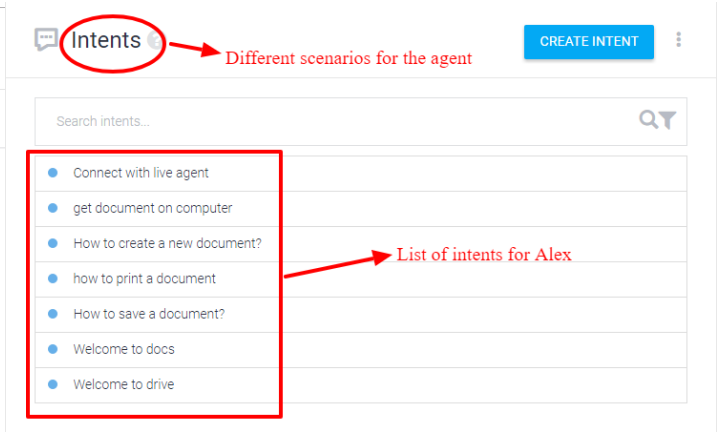


Figure 2: Creation of Rules for AI agent

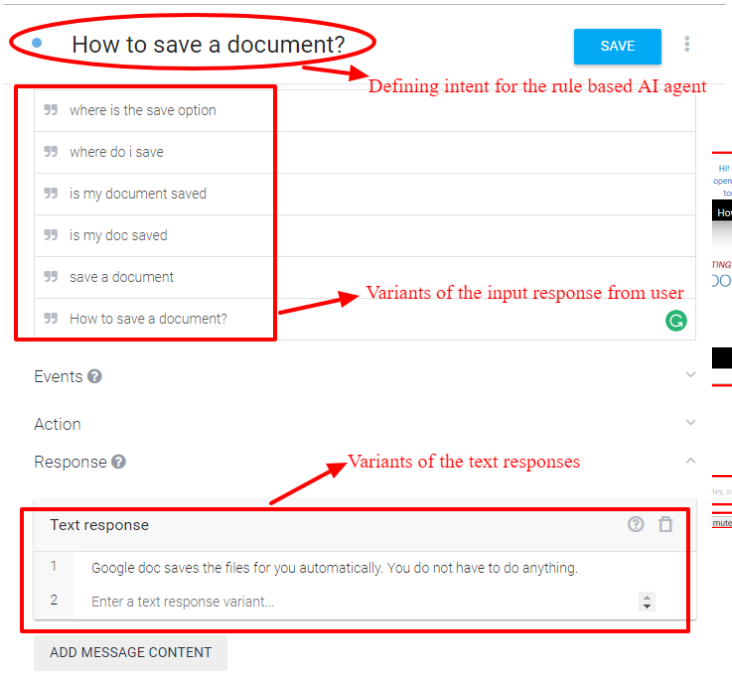


Figure 3: Creating Rule for Downloading a Document

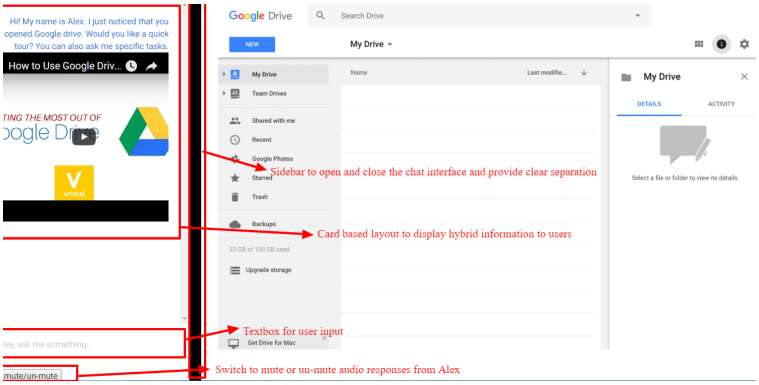


Figure 4: KnowHow on main screen of Google drive

FINAL PROTOCOL

Review the outcomes of the last session and the proposal, particularly the problem being addressed and ensure you're in agreement on these points with the participant. If not, make note of any misunderstandings because these may inform the next project deliverable.

Prior Outcomes

- The participant wanted the system to be based on her actions
- The participant preferred the sidebar design
- The user wanted to learn more about google drive, and specifically google docs.
- PRIOR system too fast!
- type size too small

Evaluate the final prototype. First have one of your teammates demonstrate the prototype to show the participant how they would interact with your design. However: Do not show your participants how to perform the three specific tasks. Instead, show them how paper prototyping works, how your system generally works, and give an example of something specific that is sufficiently different from your tasks. Then, ask the participant to complete all three tasks.

Tasks 8:46

0) Explore and understand Google Drive → easily completed
1) Create a Google Document → submenu confused user
2) Edit a Google Document
3) Learn how saving works with a Google Document
4) Learn to download/print a Google Document → user attempted to print w/o asking system

participant was able to hit enter after pause.
(looked for a "send" button)

Make a log of critical incidents (both positive and negative events). For example, the person might make a mistake or might comment on something they like. Write it down along with a description of what was going on. Later you should discuss and prioritize these events and assign **severity ratings** to problems. These ratings should range from 0 for no problem to 4 for usability catastrophe.

- print 2/4 (able to ask system)
- no send button, participant did not notice "hey, ask me something" at first

Close to what had expected.

Could use IRL "is this available w/ other g-suite products?" "I want to learn g-slides next"

Conduct an open-ended discussion with the participant about the designs.

- Were these designs overall similar to what they'd been expecting?
- What OVERALL aspects of the designs did the participant like and dislike?
- How does the participant envision an improved ~~final design? (we will elaborate with a sketching session)~~

Likes:
- good, easy to use.
- liked the ~~text~~ voice version, thought that was useful.
- good type size! / seemed separate from "google"

Dislike:
- "send" button missing
- "Hey ask me something" not visible enough. "maybe it could light up or something?"
- same color as google?

4! — giving 4 because want to interact back via voice!, and really want a "send" button

Appendix E

List of critical incidents identified for each prototype during evaluation

For the low-fidelity prototype:

Design One:

User could not find the sidebar

The lack of functionality was confusing

Speed of mock chat was confusing

Design Two:

User did not understand dropdown options

Functionality of rollover was confusing

Was confused when looking from the right side of the screen to the left

For high fidelity prototype:

The user wanted an explicit send button in the chatbot interface. She found it confusing as to how to proceed further after typing the question. She could not discover that pressing the "enter" key would do the purpose of a send button.

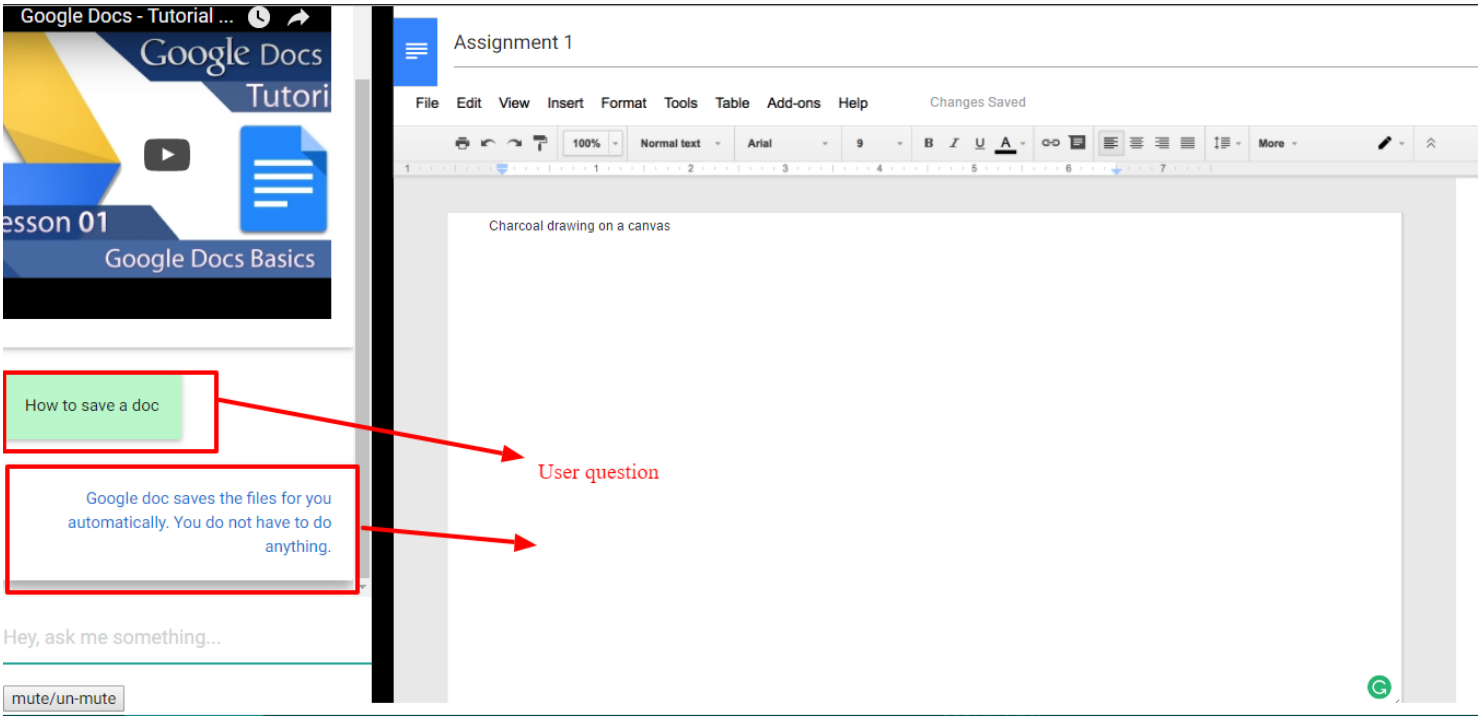
The user also found the prompt text to be hardly visible which made it difficult to identify the textbox.

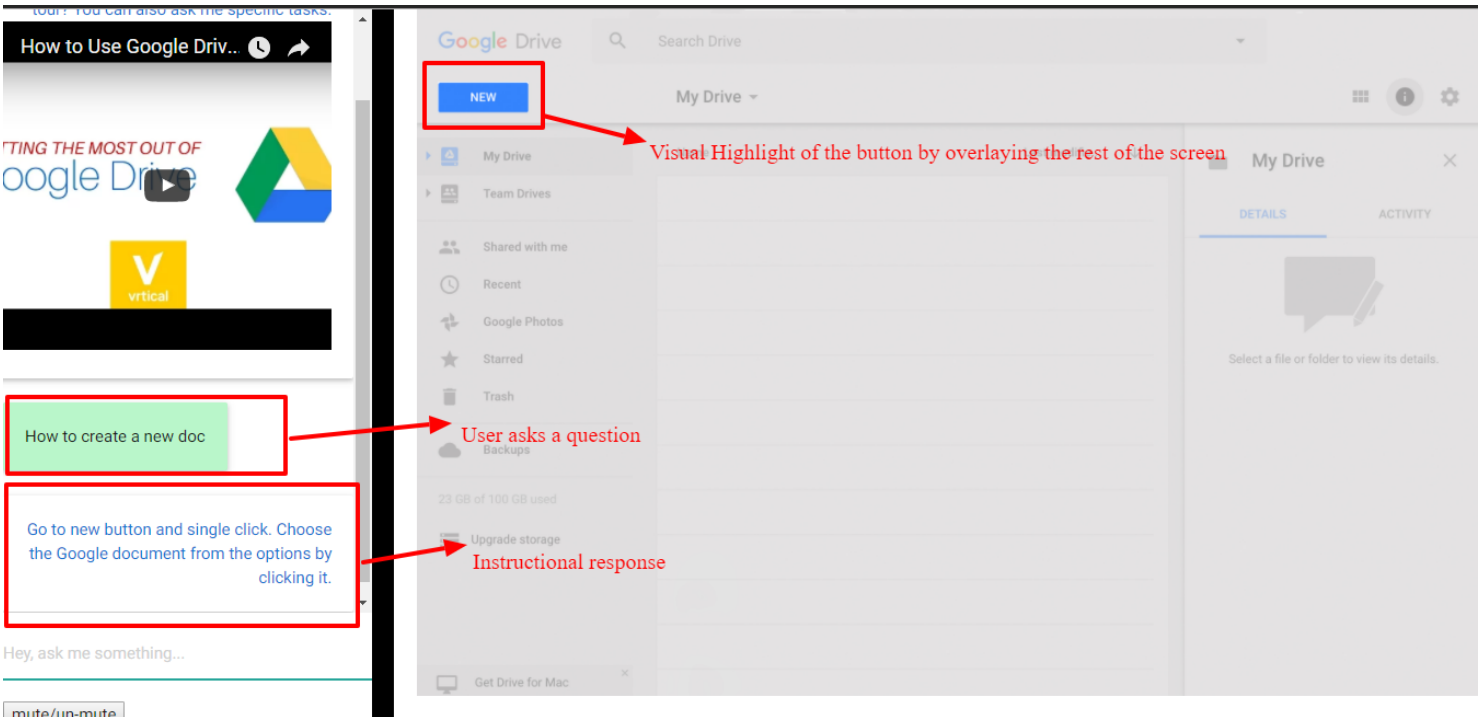
The colors used in the chatbot interface needed to be very different from the colors used in the web application so that the user does not feel confused about the help system and can separate it from the actual web application.

User particularly liked the speech response that the chatbot provided. She found it to be natural and more human-like.

Appendix F

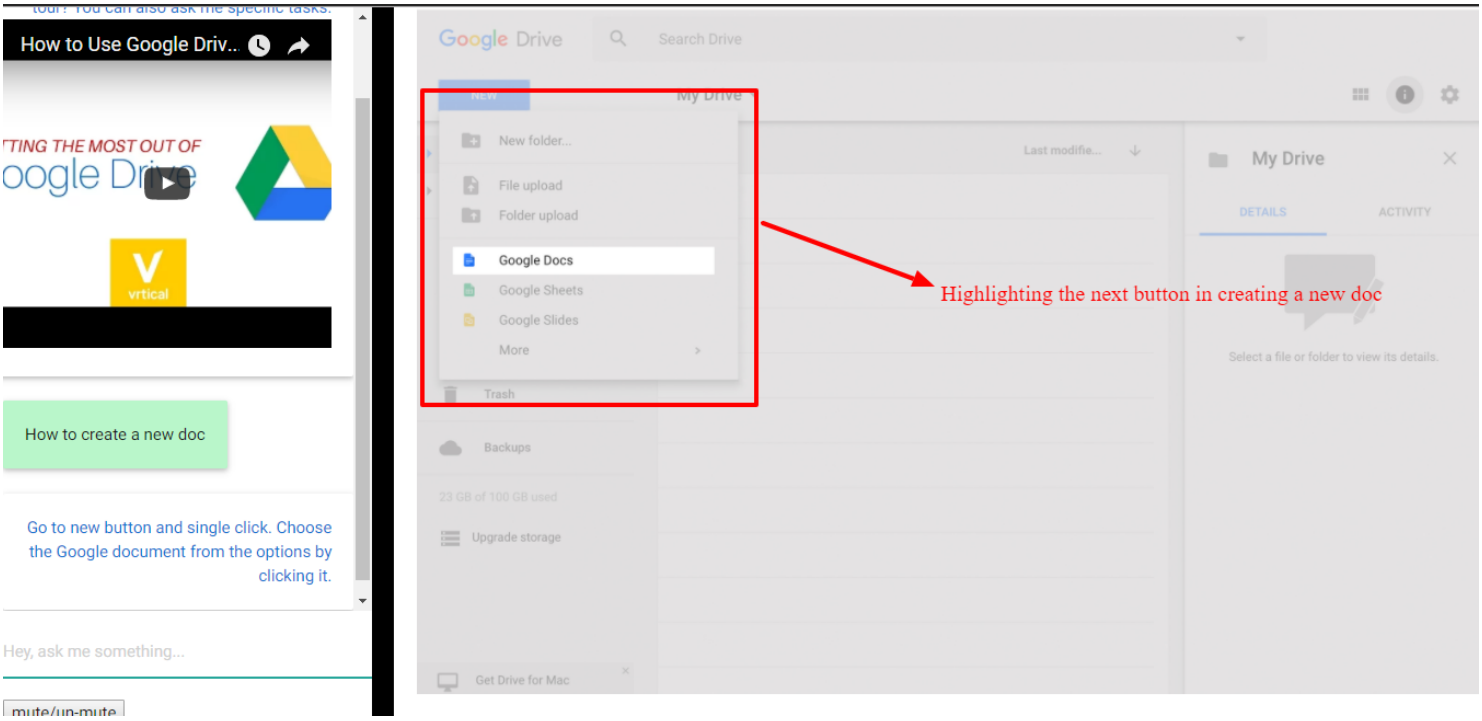
Screenshots of High-Fidelity Prototype

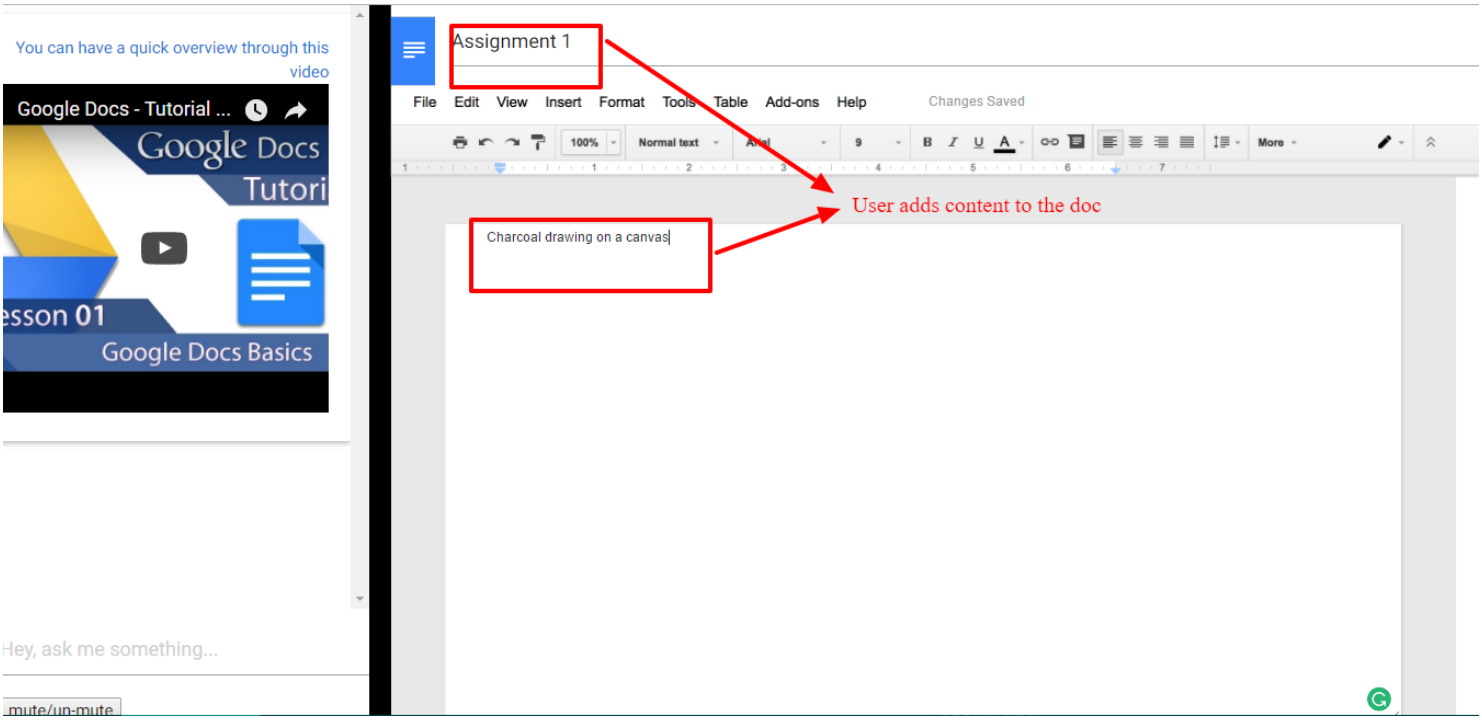




Appendix F

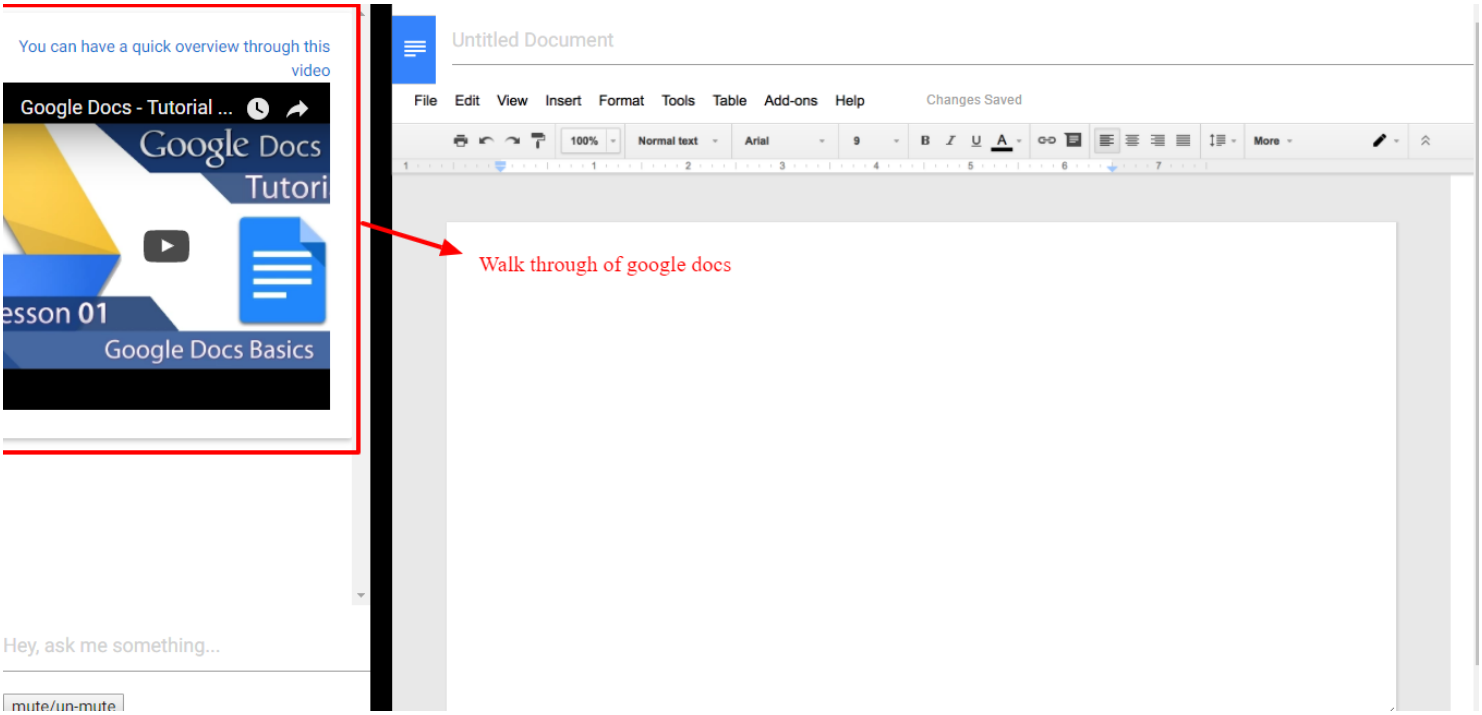
Screenshots of High-Fidelity Prototype

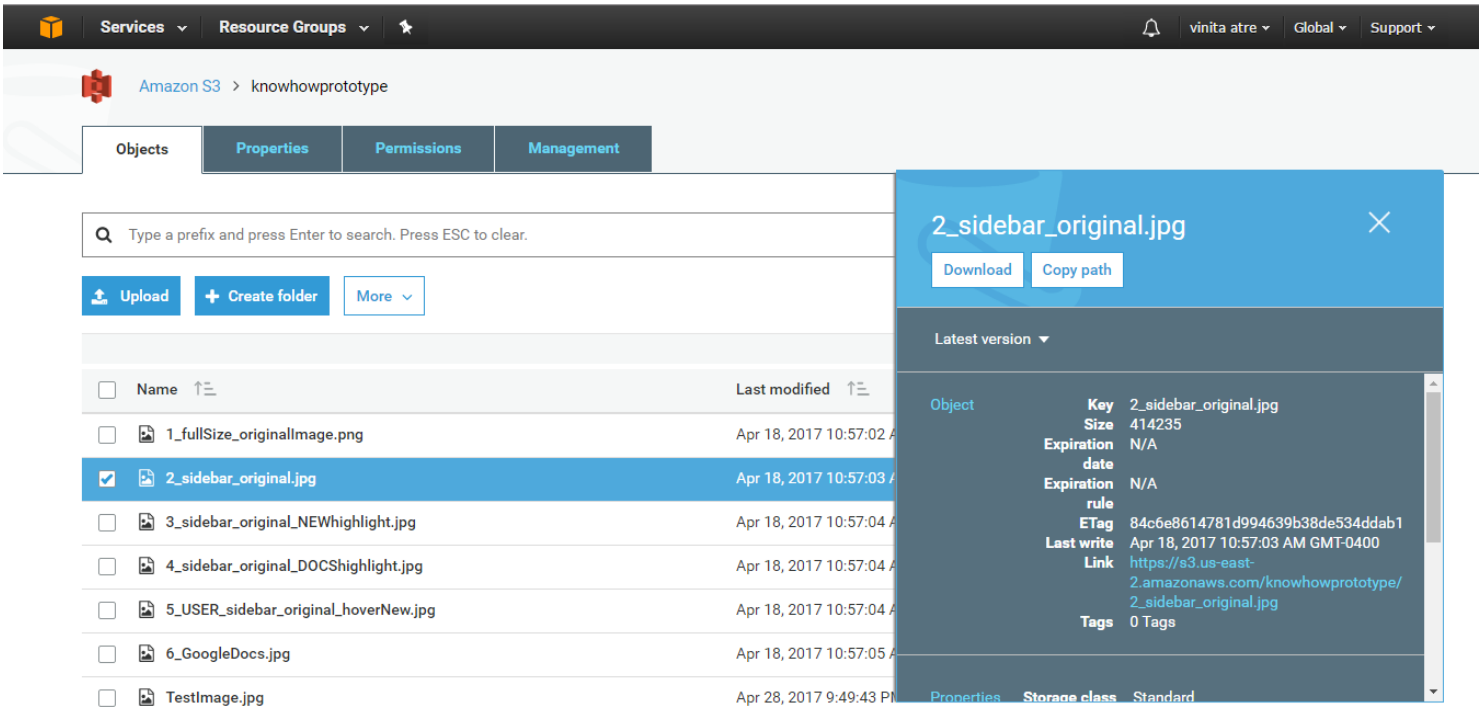




Appendix F

Screenshots of High-Fidelity Prototype





Appendix G

Code and Video links

GitHub Code Link:

https://github.com/vinita-atre/KnowHow--ChatBot-for-learning-to-use-websites-prototype-/tree/master

Demo Video Link:

https://www.youtube.com/watch?v=Wxk4TugRfow&feature=youtu.be