**Chatbot Project Report – COSC 310 Assignment 2**

Team 31
Mohammed Al-Surkhi
Jordan Colledge
Gabriel McLachlan
Jordan Ribbink
Nathan Wright

**Project Description:**

The project consists of a chatbot built in Electron; the chatbot comes with a generic visual interface to ensure simplicity of use and understanding. (Apparently this is extra credit for the next assignment, which we weren't aware of until it had been implemented.) The chatbot takes on the role of a doctor, who can be asked questions about different symptoms and describe the likely illness and remedy. Thus, the user takes on the role of a patient.

GitHub repository can be found at: https://github.com/cosc310-project/chatbot-app

**Our Github Usernames:**

Mohammed – msurkhi-1106
Jordan C. – ItsMyFuneral
Gabriel – GMcLachlan45
Jordan R. – jribbink
Nathan – DapperShark1

**Software Development Lifecycle (SDLC): Waterfall**

Rationale:

With a smaller project such as this one, we found it reasonable that we would be able to use the waterfall model. It makes sense, as each assignment (assignment 2, assignment 3…) is separated, but complete, paralleling Waterfall's separation into versions.

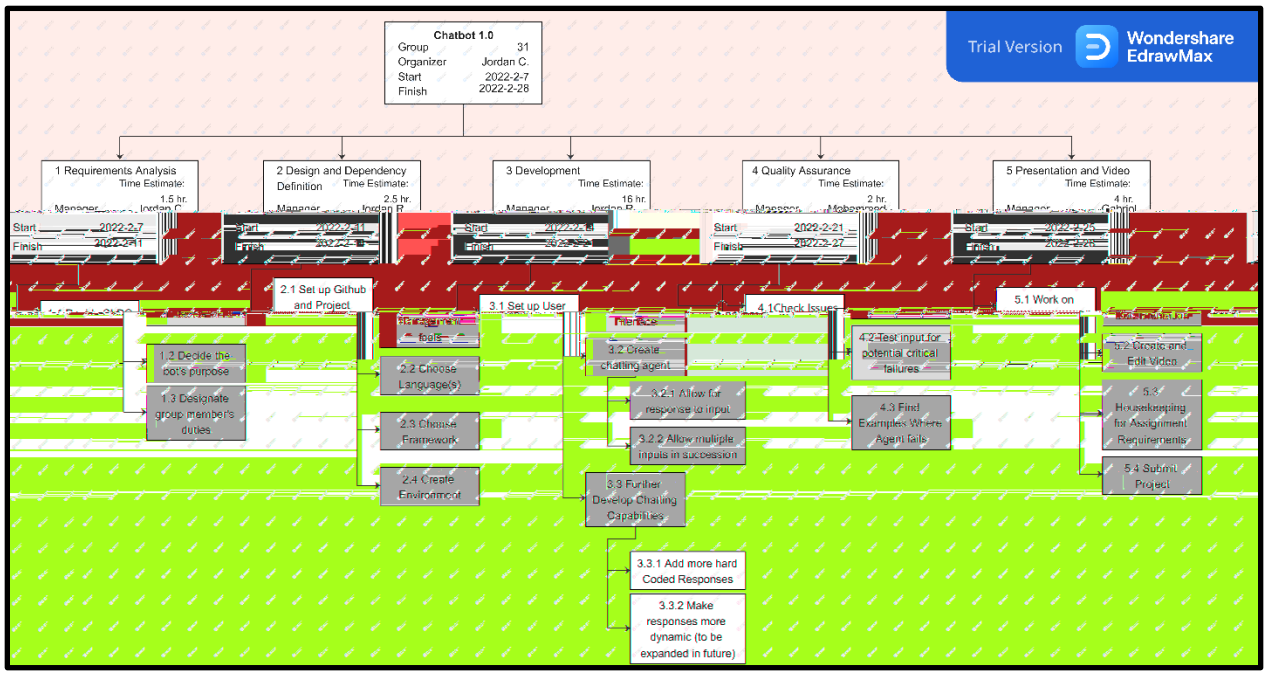
The main downside is that it's hard to accommodate change once a phase is finished in waterfall, but as the project is on a relatively small scale, we agreed that we would be able to finish the phases without much difficulty. Plus, since this project is relatively low stakes, if we run into any cataclysmic issues that *must* be addressed, we can allow ourselves to go back. Any minor issues will be saved for version 2.0 though.

**SDLC Phases:**

- Establish requirements
    - Determine programming language and IDE
    - Determine roles of user and agent
    - Determine topic of discussion
    - Establish and set up GitHub repository
        - Add members to GitHub repository
        - Create GitHub project
        - Add tasks to GitHub project

- System and software design
    - Study requirement specifications (from prior phase)
    - Specify system requirements
    - Define system architecture and tools
- Implementation and Unit Testing
    - Begin development in units
    - Test units
    - Fix units
    - Repeat 1-3 until units are finished
- Integration and Testing
    - Integrate units into main system
        - Push units to GitHub
        - Ensure no conflicts in integration
    - Ensure team members all up to date
    - Test system for faults/failures

**Work Breakdown Structure (WBS):**



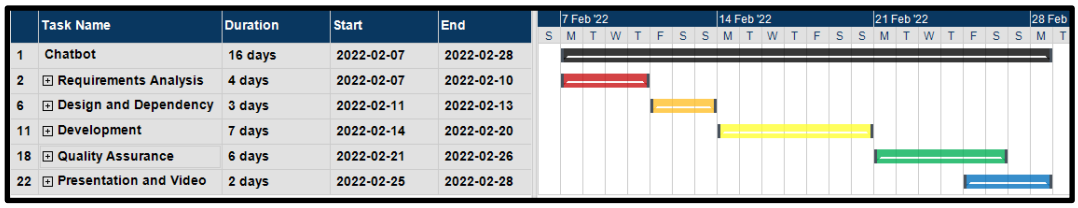

This WBS was created during the Requirements Analysis phase after we looked at the strengths from the first assignment. The main project organizer has been Jordan C., he keeps us working and was a great mediator for figuring out the rest of the tasks. Every other section was managed by someone who did well in assignment 1 in that area, so as organizer Jordan C. also facilitated the requirements, Jorden R. was the primary software architect, Mohammed focused on finding the issues, and Gabriel worked on the presentation.

You probably notice that Nathan wasn't a section organizer, but that's because he worked hard in every stage of development. We all contributed to each part a little bit but mostly focused on our portion, while he took it upon himself to go above and beyond in every part.

As this WBS was made at the start of the project, it doesn't contain the actual time it took for each section. So, here's a breakdown of the Estimated Time vs. Actual Time

| Stage | Estimate (hours) | Actual (hours) | Reason |
|---|---|---|---|
| Requirements | 1.5 | 1 - 2(including graphs) | Planning things out was easy and seeing the strengths of each member during assignment 1, designating each role was simple. The charts took longer than the deliberation. |
| Design | 2.5 | ~4 | Github and choosing the to use Electron didn't take too long, but setting up the environment took some trial and error |
| Development | 16 | 5.5 | This section was really overestimated, mostly because the UI and bot responses was actually very simple to implement. Stringing together inputs was much easier than first expected as well, which significantly reduced the development time.<br>We decided to focus on a more hardcoded approach and would've used more time had we focused on an API in this section. |
| QA | 2 | 3 | No huge issues needing attention were found, but small typos and such were found throughout the week of QA, which added a bit of extra time. |
| Project Roundup (vid) | 4 | ~4.5 | This one was one of the most accurate. The video is simple enough, and it's just making sure we have everything in this document. |

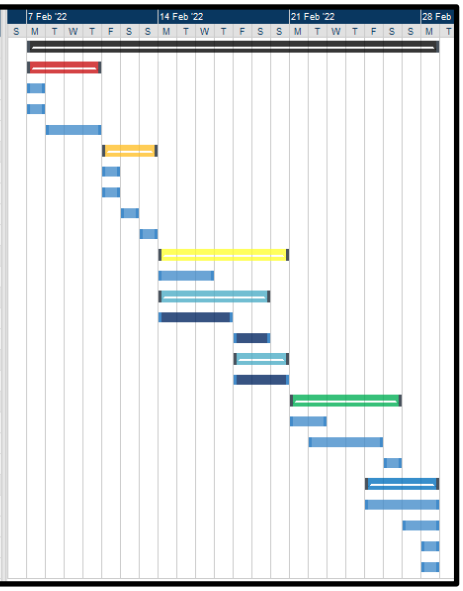
**Gantt chart:**
**(Condensed)**




This is the condensed Gantt chart for our project. It's rather simple to follow, each color is a different section of the project, and it parallels the requirements of the waterfall methodology, where all the software development parts come one after another, with no overlap. The end dates of each task are different than the WBS, but that was merely a limitation of the software.

**(Full)**

| | Task Name | Duration | Start | End |
|---|---|---|---|---|
| 1 | Chatbot | 16 days | 2022-02-07 | 2022-02-28 |
| 2 | ⊟ Requirements Analysis | 4 days | 2022-02-07 | 2022-02-10 |
| 3 | Choose SDLC | 1 day? | 2022-02-07 | 2022-02-07 |
| 4 | Designate the Bot | 1 day? | 2022-02-07 | 2022-02-07 |
| 5 | Group Member Roll Call | 3 days | 2022-02-08 | 2022-02-10 |
| 6 | ⊟ Design and Dependency | 3 days | 2022-02-11 | 2022-02-13 |
| 7 | Set up Project Management | 1 day | 2022-02-11 | 2022-02-11 |
| 8 | Choose Languages | 1 day | 2022-02-11 | 2022-02-11 |
| 9 | Choose Frameworks | 1 day | 2022-02-12 | 2022-02-12 |
| 10 | Create environment | 1 day | 2022-02-13 | 2022-02-13 |
| 11 | ⊟ Development | 7 days | 2022-02-14 | 2022-02-20 |
| 12 | Set up UI | 3 days | 2022-02-14 | 2022-02-16 |
| 13 | ⊟ Develop Agent | 6 days | 2022-02-14 | 2022-02-19 |
| 14 | Input/Output | 4 days | 2022-02-14 | 2022-02-17 |
| 15 | Allow for >1 input | 2 days | 2022-02-18 | 2022-02-19 |
| 16 | ⊟ Flesh out Responses | 3 days | 2022-02-18 | 2022-02-20 |
| 17 | Add Hard Coded Responses | 3 days | 2022-02-18 | 2022-02-20 |
| 18 | ⊟ Quality Assurance | 6 days | 2022-02-21 | 2022-02-26 |
| 19 | Check Git Issues | 2 days | 2022-02-21 | 2022-02-22 |
| 20 | Check Critical Failures | 4 days | 2022-02-22 | 2022-02-25 |
| 21 | Get Examples where agent fails | 1 day? | 2022-02-26 | 2022-02-26 |
| 22 | ⊟ Presentation and Video | 2 days | 2022-02-25 | 2022-02-28 |
| 23 | Documentation | 2 days | 2022-02-25 | 2022-02-28 |
| 24 | Create and Edit video | 1 day? | 2022-02-27 | 2022-02-28 |
| 25 | Housekeeping | 1 day? | 2022-02-28 | 2022-02-28 |
| 26 | Submit | 1 day? | 2022-02-28 | 2022-02-28 |




This is the full Gantt chart. Each of the supertasks retain their colors from the condensed version, and the subtasks are the lighter blue sections beneath.

The chart is ordered by dependency, so any tasks that fully come to the right of any other task require the other tasks to be finished (i.e., Roll Call needed the SDLC and bot designation to be finished before it could be worked on).

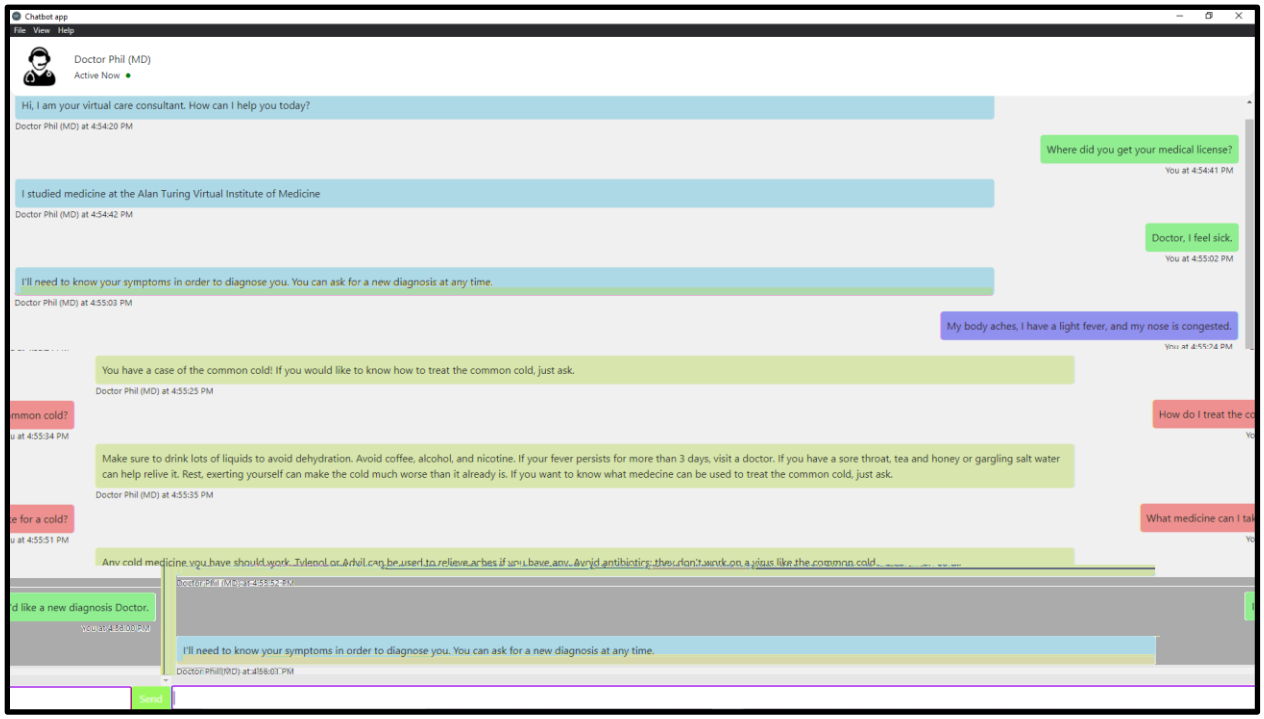
Any tasks that overlap the same time could have been taken care of in parallel (like UI and I/O), though in practice, they were still primarily worked on sequentially in order of end date.

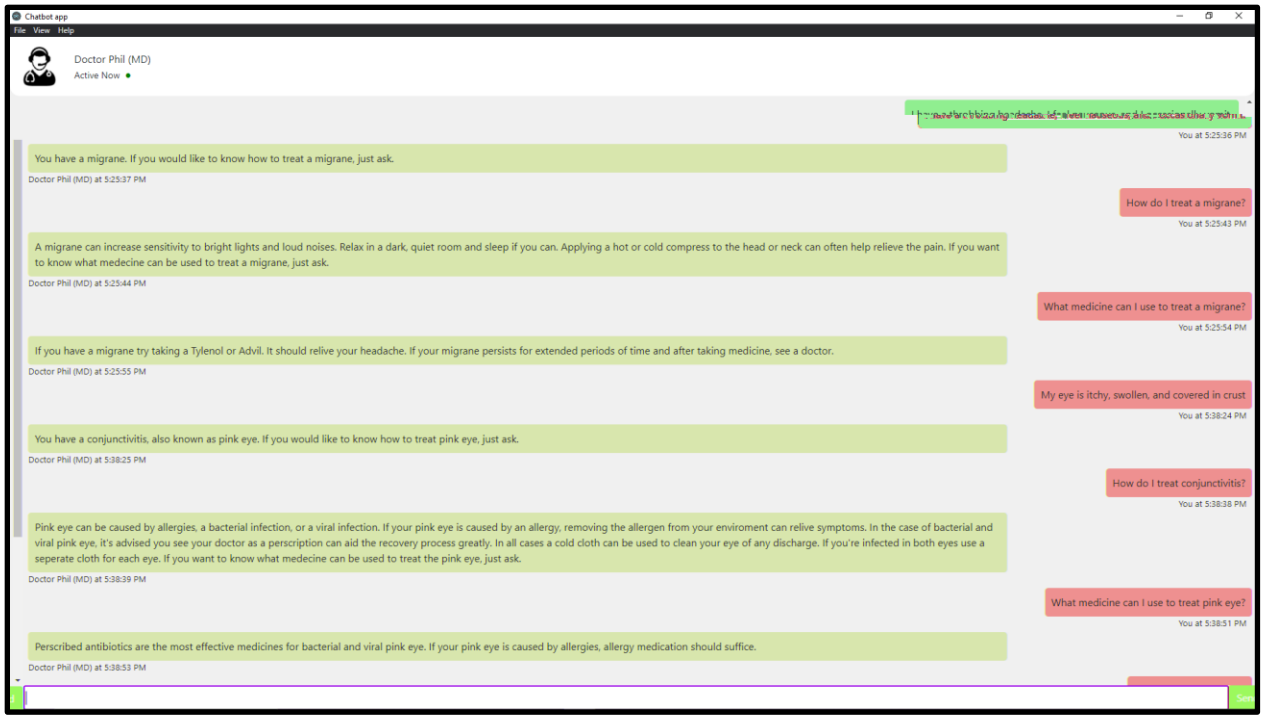
**Program Limitations:**

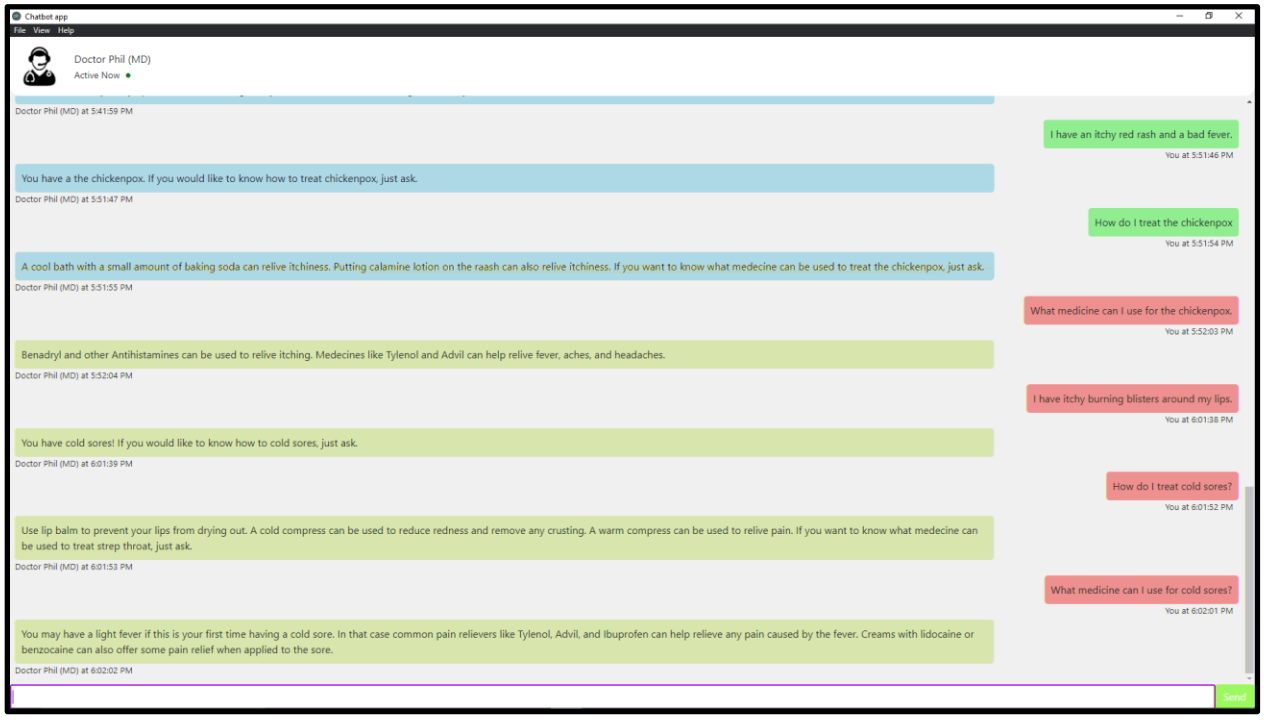
- Cannot understand typos or different terms than what's given
- Responds to strings of gibberish that include a single
- Can’t handle inputs with more than one symptom or question.
- Is restricted to its relatively small ailment dictionary (i.e., doesn’t cover every illness)

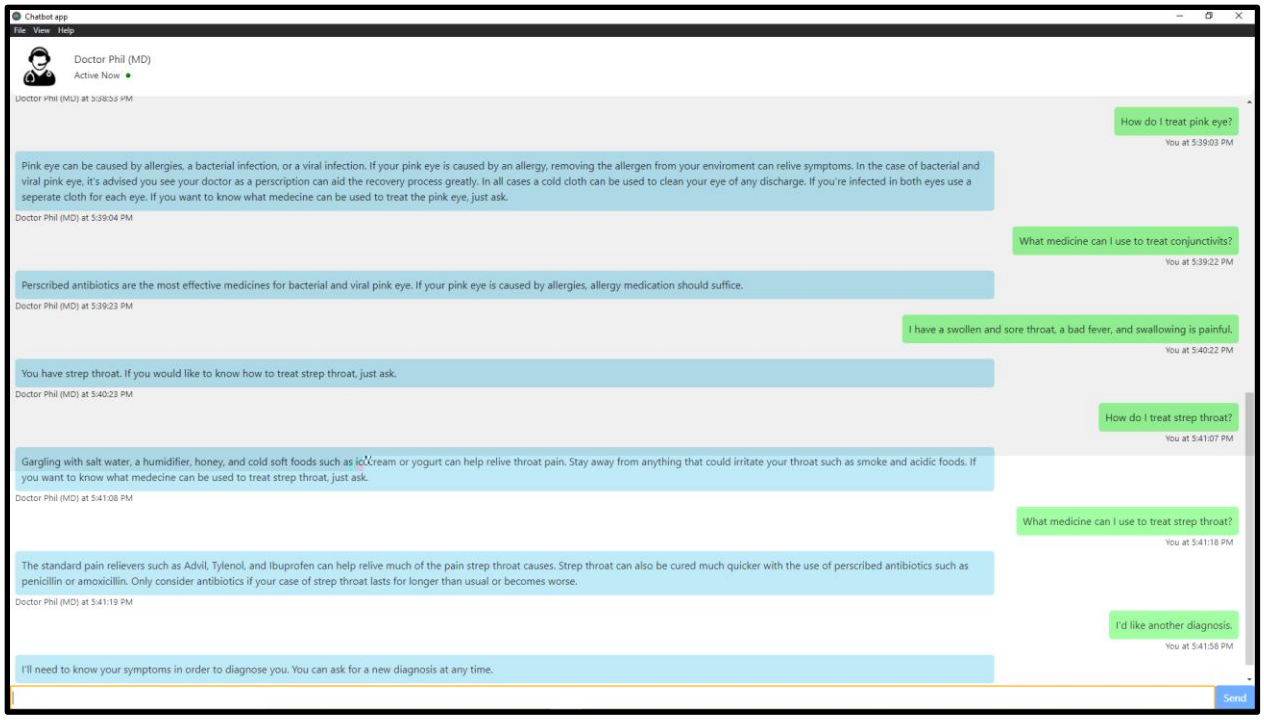
**Sample Output:**

Here's an example of a 30-turn conversation with our chat bot.

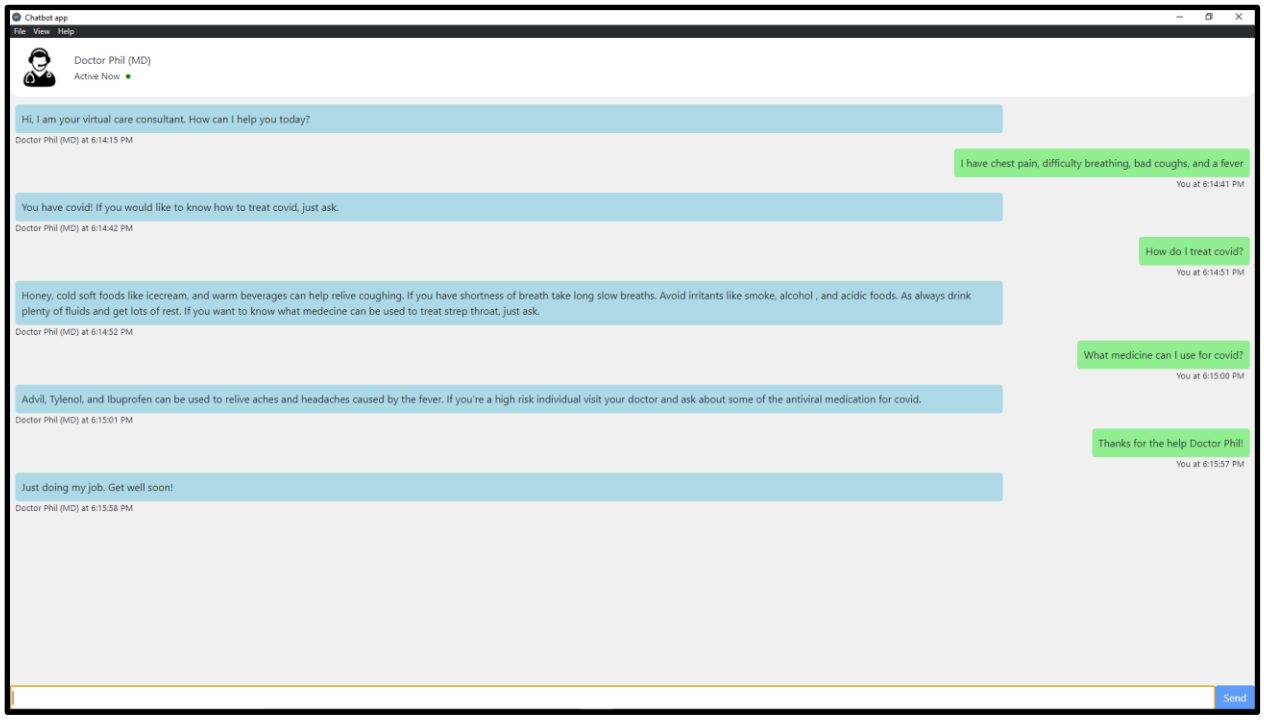




Chatbot app
File View Help
Doctor Phil (MD)
Active Now
You at 5:25:36 PM
You have a migrane. If you would like to know how to treat a migrane, just ask.
Doctor Phil (MD) at 5:25:37 PM
How do I treat a migrane?
You at 5:25:43 PM
A migrane can increase sensitivity to bright lights and loud noises. Relax in a dark, quiet room and sleep if you can. Applying a hot or cold compress to the head or neck can often help relieve the pain. If you want to know what medecine can be used to treat a migrane, just ask.
Doctor Phil (MD) at 5:25:44 PM
What medicine can I use to treat a migrane?
You at 5:25:54 PM
If you have a migrane try taking a Tylenol or Advil. It should relive your headache. If your migrane persists for extended periods of time and after taking medicine, see a doctor.
Doctor Phil (MD) at 5:25:55 PM
My eye is itchy, swollen, and covered in crust
You at 5:38:24 PM
You have a conjunctivitis, also known as pink eye. If you would like to know how to treat pink eye, just ask.
Doctor Phil (MD) at 5:38:25 PM
How do I treat conjunctivitis?
You at 5:38:38 PM
Pink eye can be caused by allergies, a bacterial infection, or a viral infection. If your pink eye is caused by an allergy, removing the allergen from your enviroment can relive symptoms. In the case of bacterial and viral pink eye, it's advised you see your doctor as a perscription can aid the recovery process greatly. In all cases a cold cloth can be used to clean your eye of any discharge. If you're infected in both eyes use a seperate cloth for each eye. If you want to know what medecine can be used to treat the pink eye, just ask.
Doctor Phil (MD) at 5:38:39 PM
What medicine can I use to treat pink eye?
You at 5:38:51 PM
Perscribed antibiotics are the most effective medicines for bacterial and viral pink eye. If your pink eye is caused by allergies, allergy medication should suffice.
Doctor Phil (MD) at 5:38:53 PM


Chatbot app
File View Help
Doctor Phil (MD)
Active Now
Doctor Phil (MD) at 5:41:59 PM
I have an itchy red rash and a bad fever.
You at 5:51:46 PM
You have a the chickenpox. If you would like to know how to treat chickenpox, just ask.
Doctor Phil (MD) at 5:51:47 PM
How do I treat the chickenpox
You at 5:51:54 PM
A cool bath with a small amount of baking soda can relive itchiness. Putting calamine lotion on the raash can also relive itchiness. If you want to know what medecine can be used to treat the chickenpox, just ask.
Doctor Phil (MD) at 5:51:55 PM
What medicine can I use for the chickenpox.
You at 5:52:03 PM
Benadryl and other Antihistamines can be used to relive itching. Medecines like Tylenol and Advil can help relive fever, aches, and headaches.
Doctor Phil (MD) at 5:52:04 PM
I have itchy burning blisters around my lips.
You at 6:01:38 PM
You have cold sores! If you would like to know how to cold sores, just ask.
Doctor Phil (MD) at 6:01:39 PM
How do I treat cold sores?
You at 6:01:52 PM
Use lip balm to prevent your lips from drying out. A cold compress can be used to reduce redness and remove any crusting. A warm compress can be used to relive pain. If you want to know what medecine can be used to treat strep throat, just ask.
Doctor Phil (MD) at 6:01:53 PM
What medicine can I use for cold sores?
You at 6:02:01 PM
You may have a light fever if this is your first time having a cold sore. In that case common pain relievers like Tylenol, Advil, and Ibuprofen can help relieve any pain caused by the fever. Creams with lidocaine or benzocaine can also offer some pain relief when applied to the sore.
Doctor Phil (MD) at 6:02:02 PM
Send

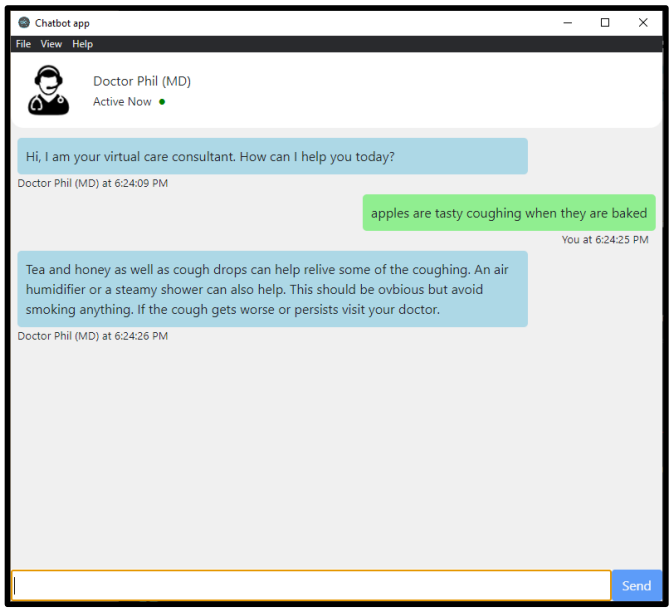



And a bonus example where we thank Doctor Phil:



**Improper Handling:**

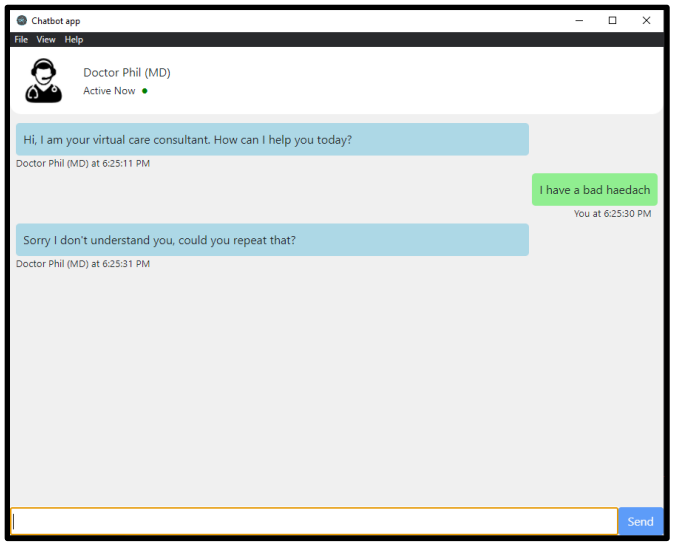
Here are a couple of examples where the bot fails to handle a conversation properly:




In this example, a weird, nonsensical input triggers a real output.

The error message should have been displayed, but instead we get a treatment to help with coughing, just because the word coughing was entered.

While it's not necessarily the worst output, it could throw doubt on the robustness of the bot and may need to be addressed.




In this example, a simple typo for an obvious word triggered the default error message.

The bot should have inferred that "headache" was intended, but instead gave us the error message.

This could be a problem if someone just didn't know how to spell their ailment, and possibly would need to be addressed.

**CASE Tools Used:**

*Github and Github Projects:* The team's decided codebase and project management system. Used to keep both the code, and the issues clear.

*Edraw Max:* We used the trial, web version of Edraw Max to develop the Work Breakdown Structure, with the various tasks and subtasks

*MatchWare by MindView:* Used to create the Gantt chart to outline the plan from the WBS. Allows for a neat and tidy view of the course of action for the project.

*VS Code:* The IDE of choice for the development of our project.