1. **Project Title:** SympChat
2. **Project Summary**: Our project consists of a medical diagnostic tool that aims to utilize Natural Language Processing to help users easily diagnose their illness symptoms by comparing them to a reference set of data stored within our database. The user will select from the body where their issue is and/or input keywords aligning with their issues (eg. Nausea, Swelling, Bruising, etc.) and using data scraped from proven sources, such as Mayo Clinic and WebMD, SympChat can quickly provide possible diagnostics and remedies. By allowing the user to save their past illness records and update with the official doctor diagnosis, future cases with similar conditions will become more accurate, and the user will effectively have access to a personal health log. This tool is aimed towards less tech savvy audiences who aren't familiar with AI tools and deep web searching. We plan to make an easy to use simple interface that will allow young children to easily figure out their cuts and scrapes and headaches if home alone. We plan on using Rake to handle the keyword extraction and processing, and React.Js for the frontend. For the heat map visualization there are variations of Python packages that we will select.
3. **Description:** Our intended audience is for either young children or older patients who do not have the necessary technical abilities to navigate complex AI tools and become buried in pages and pages of websites. By offering a clear visual model where users can easily mark their pain and easily type their issues, the minimalistic UI/UX will cater to less tech-savvy individuals. For example, an 8 year old who is home alone cut his arm and now it's bleeding. He does not understand how to use ChatGPT or search up his issue on WebMD. Instead, he clicks the visual of the arm and types in blood or cut. SympChat can handle the grunt work and pull from the database various solutions to help his pain. This will be saved in the system as well, so next time, if he makes the same mistake and forgets the steps toward treatment, he can click his profile picture and review the steps. Later on, his parents can access the chat and provide the doctor with an update to enhance the accuracy of the result, so next time, it helps their child even better. The potential to include images in the visualization will allow users to compare their physical symptoms with an example. An easy to use, least-typing-required interface will allow very elderly individuals and very young individuals to gain access to thousands of common health issue cures.
4. **Creative modification:** The creative modification our team will be doing for our project will be an interactive symptom heatmap visualization of the human body where users can click on affected areas and the system will dynamically adjust symptom probabilities and possible diagnoses based on what was marked as affected. It will improve the functionality and user experience of our application as the unique visualization concept makes the whole process more interactive and engaging for the user, differentiates us from any other existing medical chatbots, and provides more nuance and accuracy for symptom diagnosis as an individual may not know exactly what part of their body is

affected but if they can just highlight where they feel pain or the symptoms it will yield more accurate results.

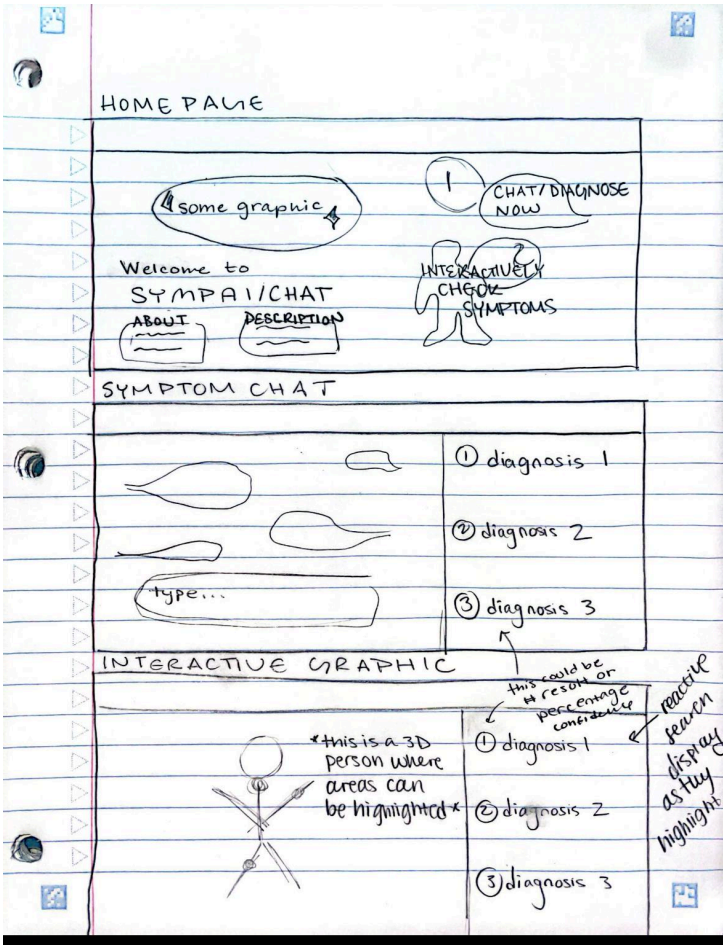
5. **Usefulness:** SympChat is a web application designed to help users assess potential illnesses based on their symptoms. The user starts by click on the area of the body where the issues are and/or inputting their symptoms into the application, and in turn gets a list of illnesses their symptoms match. Using data from sources like the Kaggle datasets, WebMD, and Mayo Clinic we can match the symptoms inputted by the user to the illnesses in our database. Additionally we will also have a place where post doctor evaluation, the user can confirm their final diagnosis as well as an interactive 3D heat visualization map where they can highlight affected areas of the body if they'd like. This will help make the application more accurate, engaging, and user friendly and open up the opportunity to report missing illnesses from the database. While similar websites, like WebMD exist, the user has to usually search up symptoms individually. With our app they can input a list of symptoms and in return be given a list of possible diseases as well as confidence level, streamlining the overall process.
6. **Realness:**
    a. https://www.kaggle.com/datasets/dhivyeshrk/diseases-and-symptoms-dataset
    b. https://snap.stanford.edu/biodata/datasets/10021/10021-D-DoMiner.html
    c. The above two datasets are what we are planning to use for the project. The first dataset is from Kaggle, and has 773 unique diseases and 377 symptoms, with approximately 1,420 disease-symptom pairs. Each entry involves a disease name and then the corresponding symptoms for the disease. The dataset is a CSV.
    d. The second dataset comes from the BioSnap collection from Stanford University. It has 9284 rows, each containing a different disease and the corresponding symptoms for the disease(2 columns). The data is in the TSV format.
    e. In the future, we may also add data to our dataset by web scraping from sites like Mayo Clinic and WebMD if we deem it necessary in the future. Also, we can pull drug treatment data from the OpenFda API or HealthGov API.
    f. For our 5 entities, we are planning doing the following: Users, Diseases, User Consultations(ie input), Medicines, and Symptoms.
7. **Functionality:**
    a. **Description:**
        i. Users can "chat" with our database retrieval/NLP bot, telling it a list of symptoms that they are experiencing as input and it searches for possible diagnoses. They will see a search result output of possible diagnoses retrieved from our database on the right hand side to summarize results as well as a more basic, chat-like response in the chat interface and the respective level of confidence for the diagnosis. ("Read" part of CRUD functionality)

ii. Users can make an account in order to save their recent searches and other regular account functionality.

iii. Users can manually add a list of symptoms and what their illness ended up being diagnosed as by a doctor inside the chat part of the website, in order to contribute to our dataset, build the SympAI community, and provide more accuracy to our application over time. They can also update this or delete it if they did it wrong the first time or the diagnoses developed or changed over time. (“Create”,“Update”, and “Delete” part of CRUD functionality as well as Search functionality by searching for symptoms)

iv. Users can use an interactive symptom heatmap visualization of the human body to click on affected areas and the system will dynamically adjust symptom probabilities and possible diagnoses based on what was marked as affected. (Creative modification)

b. **UI mockup:**

i. 

c. **Project work distribution:**

i. We will split up the tasks evenly, all looking to do a portion of the project in order to maximize learning and collaboration.

ii. On the frontend, Helena will do the home page, Aman will do the interactive graphic, Aarushi will do the chat page, and Michael will do the diagnosis sidebar that appears on the chat page and interactive graphic page.

iii. On the backend, Michael and Aman will do the NLP functionality to ensure accuracy on user input, Aarushi and Helena will do the CRUD operation code, and all of us will work on the database set up together in order to gain understanding about this as we are in 411: Database Systems.