

1. Describe what data is stored in the database. (Where is the data from, and what attributes and information would be stored?)

We chose the “Chronic Illness: symptoms, treatments and triggers” dataset provided to us as one of the example datasets. This dataset is from a company called Flaredown, and contains a series of health attributes that users voluntarily fill out. Flaredown helps users understand how exposure to certain treatments and environmental factors impacts their pain levels for the ailments each consumer is facing.

Our team would store the user_id, age, sex, country, trackable_id, trackable_type, trackable_name, trackable_value, and a series of attributes that we build based on the existing attributes in the dataset. This includes the expansion of the current “multi class” trackable_type attribute to a series of binary attributes that each individually describe a consumer’s use of a specific trackable_type. For example, if a user with user_id 1 enters a trackable_type “Condition”, our attribute “Condition” for the user would have a “1” written to it to represent “True”.

2. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?)

- a. Users on this website will be able to input their demographic information, as well as search for symptoms from our dataset that match the symptoms they are experiencing.
 - Search bar for symptoms to match the existing symptoms from dataset
 - May select as many symptoms as desired and display all symptoms as added (and may remove them)
 - Input demographic information using drop down menus
- b. We will use a best match algorithm to find the closest conditions that match the inputted symptoms in order of closeness. User will be able to click the condition and route to the page which will contain:
 - Demographic chart for the gender of people with this condition
 - Top symptoms and treatments given with this condition
 - Other charts from analysis
 - Our complex map
 - External link to medical site with more information (below)
 - will call upon a [National Library of Medicine API](#) to provide links to the symptoms and description that an individual may be experiencing.
- c. The general user flow for our app is:

- User will enter their Age, Gender, Location, Personally Identifiable Information Data
 - Then, they will choose which symptoms they are experiencing.
 - A list of possible conditions they may be having (with a % likelihood) is displayed, along with the following information:
 - The treatments that people with each of these conditions have
 - The top symptoms that people with this condition have
 - The top tags/weather that people mark may trigger their symptoms
 - A link to a source that has descriptions of further information
- d. Users can also make an account with demographic information so they can log in and track their data. After perusing for possible conditions, the user will be able to have a profile that stores, adds, and updates demographic information, their symptoms, their treatments and their conditions.
- Insert conditions, symptoms, treatments to their page
 - Delete conditions, symptoms, treatments from their page
 - Update profile information

3. What would be a good creative component (function) that can improve the functionality of your application? (What is something cool that you want to include? How are you planning to achieve it?)

Since the dataset includes the patients' countries, we can have a heat map or color map of the world where the user can select a symptom and the countries are colored according to the number of patients.

This can be achieved with a query over specific countries and the number of patients with the query symptom and map them to the UI, probably with a map API.

4. Project Title

Simplify

5. Project Summary: It should be a 1-2 paragraph description of what your project is.

The goal of our project is to help people find conditions they may have based on the data, as well as store and keep track of their own conditions, symptoms, and treatments. Our project will deduce a user's most likely condition from a set of

symptoms and provide information relating to the condition. User's will also learn about demographic information about their diseases and how they compare. With our creative feature, we will create a heat map on a world map to find which diseases are most popular in the world and display some information, so people can learn more about how their location may impact conditions.

6. Description of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?

Many people with medical symptoms may not be able to afford regular doctor visits, or they may want to do more research on their own. To this end, the app assists them by presenting other people's past experiences of people with similar conditions: the conditions they experienced, the treatment they received, and other relevant data such as age group and location. The user can then use this information to complement their treatment, or to seek further medical help.

Moreover, many conditions and illnesses are not well documented on the internet. In many cases, new illnesses have little data, and the only way individuals can learn more about their symptoms is through the experiences of others in the same boat. Symplify assists people who are facing newer illnesses to find a community of members that are going through the same symptoms.

Overall, Symplify's goal is to create communities based on an illness for people to discuss their ailments on. We believe that through the power of shared knowledge, the community can come up with solutions that fix other people's problems. Think about this as a StackOverflow for medical illnesses.

7. Usefulness. Explain as clearly as possible why your chosen application is useful. Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?

This application is useful because it reinforces the interaction between users regarding chronic illness discussion and offers users more information regarding potential ailments that they may be facing. Users are able to search for an illness or diagnosis based on symptoms they are experiencing. They can also learn about treatments of other people who share similar attributes and who might have similar symptoms and illnesses.

Moreover, users are also able to share their own background, symptoms, treatments and triggers, which provides references for others and increases the shared knowledge in the community.

In terms of our competitors, we identified [WebMD](#) as a major player in the industry. WebMD has an application called “Symptom Checker”, which allows users to input their background information and symptoms to determine their treatments. However, this website is different from ours. Besides the basic functions WebMD offers, we also allow users to share their own information, including symptoms, treatments, and even unexpected environmental factors, such as weather and tag. We further distinguish from WebMD because we allow users to upload symptoms, as opposed to scraping symptom data off the internet.

8. Realness. Describe what your data is and where you will get it.

Our dataset includes important information about the diagnostic process such as Symptoms, Conditions, and environmental factors called “Tags”. These data types, along with a couple other highly applicable categories, will help Symplify classify the types of ailments that our users are facing, along with a link as to where our users can read more about their ailment. Our data is taken from a Flaredown app, which stores user submitted conditions, symptoms and treatments. We will get it from the following source:

<https://www.kaggle.com/datasets/flaredown/flaredown-autoimmune-symptom-tracker>

Here is some more detailed information about our attributes:

- User: includes an ID, age, sex, and country.
- Condition: an illness or diagnosis
- Symptom: self-explanatory
- Treatment: anything a patient uses to improve their symptoms, along with an optional dose, which is a string that describes how much they took during the day
- Tag: a string representing an environmental factor that does not occur every day
- Food: food items were seeded from the publicly-available USDA food database
- Weather
- HBI: severity index

9. Description of the functionality that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e., things that one could create, delete, update, or search for). Read the requirements for stages 4 and 5 to see what other functionalities you want to provide to the users. You should include:

1. **A low-fidelity UI mockup: What do you imagine your final application's interface might look like? A PowerPoint slide or a pencil sketch on a piece of paper works!**

We drew up a mobile implementation, but the **design remains the same for a website implementation.**

See Lofi Mockup Below↓↓↓

Simplify

< **Personal Info** Exit

What is your age ?

Ex:

What country are you in ?

Ex:

What is your gender ?

Ex:

What is your phone number ?

Ex:

Exit Next

Simplify

< **Symptoms** Exit

Headache

Headache when running

Head always hurting

Headache behind eyes

Headache and nausea

Current Symptoms

Stress Irritation

Joint Pain Limping

Fatigue Sore Throat

Exit Next

Simplify

< **Conditions** Exit

① Anxiety

② Depression

③ Cancer

④ Gastroenteritis

see more

Simplify

< **Anxiety** Exit

Symptoms **Treatments**

•

•

•

•

Anxiety: By The Numbers

Join Community!

Learn More

LoFi Design

Simplify:

2. **Project work distribution: Who would be responsible for each of the tasks or subtasks?**

List of the person responsible for which exact functionalities in section 6. Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

We believe that a project of this nature should be split by an industry standard workflow separation framework: *frontend* & *backend*. We have further identified subtasks that need to be completed for this project to succeed. We envision these two arms (frontend, backend) functioning more or less concurrently throughout the course of the semester using an Agile setup for task delegation.

GENERAL RESPONSIBILITIES

1. Frontend Functionality
 1. Design (Figma): **Effie and Faraz**
 2. Implementation (React) **Effie and Faraz**
 3. Data Visualization **Runlin**
2. Backend Functionality
 1. Route **Minh and Runlin**
 2. API / DB Modification **Minh and Faraz**
 3. Database Query / Info Retrieval **Runlin and Minh**
 4. OAuth / Middleware **Effie and Runlin**
 5. User data (insert/delete/update) on profile **Minh and Effie**
 6. Visualization Data Endpoints **Faraz and Effie**