# Lilly Technical Challenge Documentation

## Approach

I started my approach working through the objective chronologically. I started first by running the program and ensuring that it was capable of running on my machine. I then made an effort to ensure that the HTML was working and to display it to the user. I then systematically worked through the objectives, displaying data, handling the errors in a user-friendly way.

I then worked on the forms to create, delete and update the medicines. I did all the backend functionality first prior to working on any styling. After I had all the forms working, I made sure to style the website in a user friendly, aesthetically pleasing manner.

I worked through the objectives in the order I did to ensure that I had one part working prior to moving on to another task, also allowing me to compartmentalize each task as I worked.

As for external resources, I used a few article tutorials regarding JSON handling through JavaScript as this is something I had not done before and wanted to grasp in order to complete the task. Overall, I would occasionally use stack overflow and miscellaneous websites to help me with the bugs I encountered throughout.

## Objectives - Innovative Solutions

I made a specific effort to work with the code I had been given and make minimal changes to the functionality in the python file. If I was working within a team on this, slightly changing the method to make my job easier could cause issues down the line as the original design may have been done that way for a specific reason, that I am not aware of as it is in a sector I am not involved in. Instead, I just added methods that suited the functionality I needed and utilised those methods in calls.

I am particularly proud of the design I implemented, as I believe it is use friendly and enjoyable to look at, a fast improvement on the original implementation. As well as this I am proud of the single pages, as I decided this was better than a drop-down menu of some kind to display a specific medicine and edit it.

## Problems Faced

I struggled with create a medicine. I had issues with the action not being sent to the JSON file. I think this is as I have not worked with JSON before, but the problem was create\_med() not being called correctly. I failed to prevent a default form submission and instead needed to explicitly send the data to the backend where create\_med() would handle the data, and send the response back.

As well as this I struggled with the delete function, as when I was originally writing the code I sent the data to delete\_med() as a query parameter, as that is something I have done coding previously. In this I then modifying delete\_med() to handle query parameters. This worked but I found it to be a clunky solution, rather than just returning the data to the backend as form data in the first place, so I reverted the code and had the frontend return the data as form data instead, which was better and still worked.

## Evaluation

I enjoyed the challenge, it was nice to be able to work in web development independent of my studies, as recently I have been focusing on this and have not had time externally. It was helpful to learn how JSON interacts with JavaScript and to use it over a SQL database. I enjoyed the styling the most and found it the most streamlined part of the task, and I found the update functionality went smoother than the delete and create functions, as it used the same principles and any errors, I encountered I was able to refer back to the bugs I encountered in the earlier stages. In terms of time, I was unable to keep proper track due to some external circumstances that caused some interruptions in my work, but I do think I went over the suggested time spent on this. I think this is likely over small issues I failed to recognise, and issues in setting up the virtual environment originally. If I were to do this again, I would love to be able to work on the invalid data handling, as I believe my solution was appropriate, but I am eager to see what other solutions were available and could be better.