Design Report

Functionally, I add images to movies that have images on the OMDB website. In order to achieve this function. I add additional OMDB library in the environment and I add it into the requirement as well. Except in searching, whenever a title of a movie exists on the web page, the poster of this movie exists as well if the poster is in the database of OMDB. Pictures are bigger than words in the page, so it is more visible to users than just having titles. The background only consists pure colour blocks, users will be more attracted by the colourful webpage rather than the white background. Also, on the home page, rather than only show random movies, I also make a recommended list by how many times the movie has been clicked by users. The movie with more clicks means they are in fashion or users want to see them. By putting those popular movies on the top will help users saving time for searching the most popular movies.

When I developed the project, I decided to follows the marking schedule of the assignment, starting from C grade. This is fastest way that I could get an MVP in the early stage. As I beginner, I find it is really important to get at one page that works. Having visual page will make me much easier to find improvements or bugs that occurs in the app, so I could change and see the results. Developing MVP also let me to think less at the beginning, because I was confused when I started because there are so many functions that need to be done. Dividing a huge task into many small tasks let me could use small pieces of time for the project. I use the repository pattern to establish the service layers. Pattern will benefit more in the future assignments when there are multiple repositories. For MVP, I saw the example COVID-19 app. I drew graphs to understand all essential functions for MVP. I did not follow the test-driven idea because I was a little unclear on what I should assert to the test result. I want to see the visual things then I could be confident that I am on the right track. According to my way. I decided to develop html and css first. Then I wrote the function of the home blueprint. Layer by layer, I establish services and the repository. When the basic home page runs. I turned back to finish all tests for repository, started the services unit test as well. Then I used the Jinja to replace the elements in the home webpage. Then I established the browse movie page. The work includes adding the hyperlink that users could click on the title to redirect to the page of movie. At the same time, I started to wrote the code at the side, including the navigation and sidebar

for recommendation. During the process, I finish the functions in the Services and Utilities. Now the link only happens from home page to movie page. At this stage, I finish the tasks for grade C.

I turned to tasks in grade B. At this stage, tasks become related. I need to decide the order of finishing tasks. Because of the example from class, I decided to authentication section first, because it is a prerequisite for comment. Also research also requests modules that users need to type in and pass parameters from pages to pages. Research function could borrow some ideas or modules from authentication. I think is really good to established the layout for upper level even though at MVP. When I typed codes for Grade B, it is much faster as I was used to the layout already. Also, the Single Responsibility principle extremely workful. Many sections when I finished, I do not need to go back anymore. Even for deeper layers, I referred back in the early stage, but what I finish most work for Grade B.

When I turned to Grade A, all I need to is to write some small functions at blueprint layers and html layer. Introduce new library and the project is done.