# Patient Web App Report

Name: Haoshan Xiong

Features

There are three main functions in this Patient Web App.

1. **CSV File Reader:** The App could read data from CSV file.
2. **Viewing:** First one is to show the details of the patients.
3. **Searching:** this application could search the patient by keywords which could be Name, ID, SSN, Passport. In the result page users could click the index to view the details of the patient.
4. **Editing:** it could edit the list content or details of the patient, including adding or deleting patient.

This Web App are constructed most on "Three Layers" architecture, Model-View-Controller (MVC) architecture, Model Java Class with all function and data, Servlet Java Class and front-end JSP file.

UML Diagram
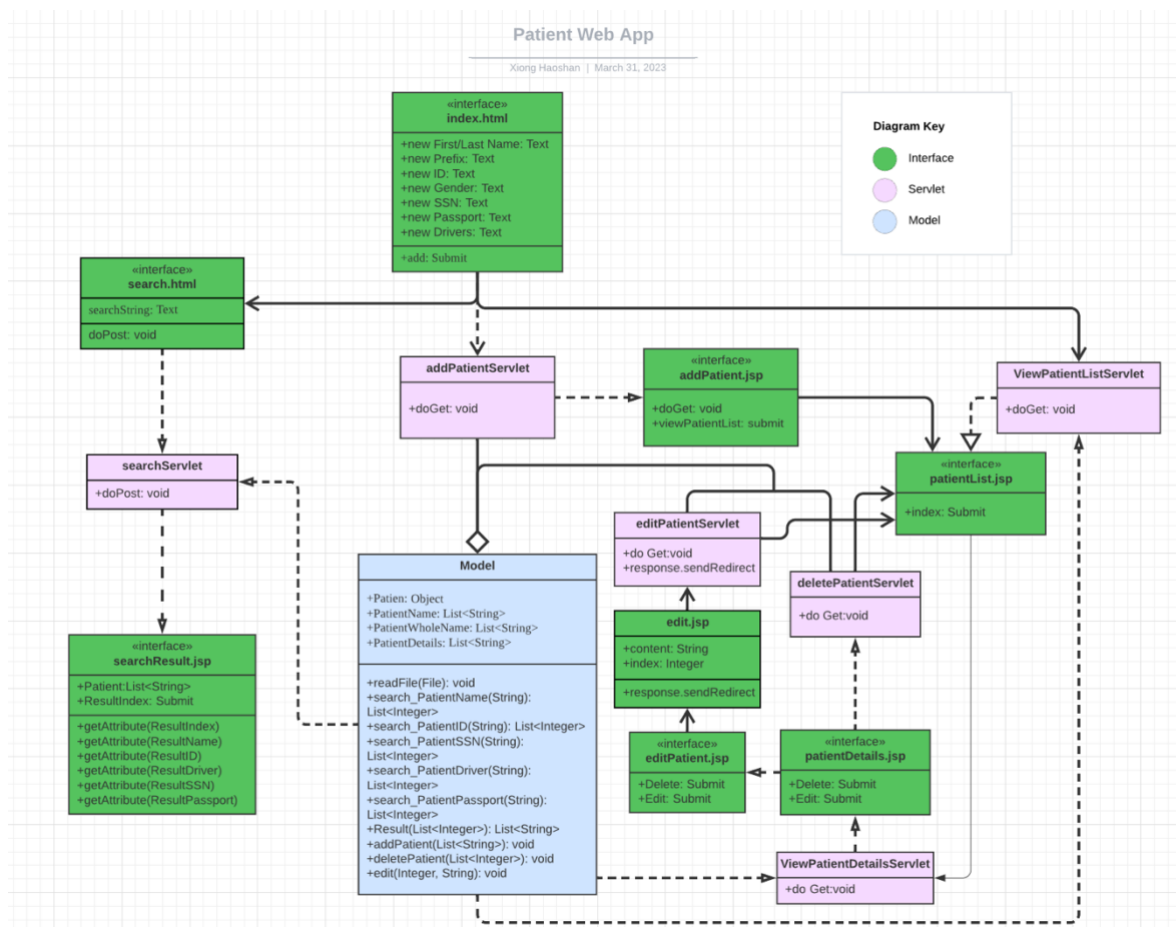
Design and Programming Process

For the Model Java Class, one of the most important roles is to read the content in the CSV file and convert it into a readable String List or Patient Sample. Each element in the List and Patient Sample would be given an index. Then, other function like "Edit", "Delete", "Search" would change the content through "index" to find the proper patient. Servlet Java Class used to analysis the behaviour of users and give a proper response by using the function in Model Java Class. JSP file is used to show the most information requested from Servlet on the website and give the concrete function that user could implement meanwhile it will also send the needed information to Servlet.

In the model, the class object "Patient" is used to save the basic and information related to search string including Name, Driver, ID, SSN, Passport. Methods in this object are used to search those related patients. Besides, there are several lists, the name of which are related to their function. For the patient Deßtails List, all information about one patient will be stored under a single index of the list and nearly all function in the project are related to this index. The methods in model are used to implement different functions, such as deleting, editing, adding, or searching which cooperate with the Patient Details list and Name list. They make the project really cohesive.

For the quality of this project, it definitely could work normally and realize most needed function, but the back code end still need to be improved, like the process the details of each patient. Using a single list is not a good way to store such a huge data.

Conclusion:
This Web App makes managing patients more convenient by giving a number of functions in different pages. The UI of this Web App is still needed to improved making the windows more beautiful.