**Design Documentation**

The main aim of the program was to create a form that accepts the results of the voting process and store this information within a database.

In problem one (1), we were asked to create an XHTML file in which it held two versions(A and B) with links to different CSS files.

Version B’s layout was then used in problem 3 when we are asked to return the information in a table structure.

The XHTML file from problem two, was used to make a form in which individuals are allowed to fill out by using the valid **\*form\*.**

**Lastly what was done was setting up the database. For this, one had to open the MAMP server which then navigates the PHP/MyAdmin which opens in a browser. To create the database, one had to import the files from the starter code given then set up all the starter code tables.**

For problems two(2)and three (3), the only difference was the JavaScript code which was commented out and to the form tag in this file the action and method was added. To the action listener the reference to the php file to be used was added and to the method, POST was added in order to allow for the more secure retrieval of data from this form.

In order to move on the problem 3,

In the php file the password, username, host and database name were first initialized. The program then tries to connect to the database using the password, username, host and database name. If the program did not get to be successfully connected, then an error would be raised. Otherise if the program was successfully connected then a message would be displayed telling the user that they have been successfully connected to the database. After being successfully connected to the database, the program takes the input from the form and checks if they have satisfied the conditions outlined in problem 2. That is if all the input fields were not empty, if all the fields except the polling station were integers and if the polling station was alphanumeric and if the total votes was equal to the sum of the votes for candidate 1, the votes for candidate 2 and the rejected votes. If any of the fields did not pass the stipulated conditions, then an error message would be displayed to the user indicating which fields were invalid.

If all the input fields passed the conditions, then the program will insert the data inputted into the corresponding fields in the StationVotes Table in the database, then executing this insertion command. After the data has been inserted into the database, the program then gets all the required data from the updated database and store it in a variable by using a connection query and selecting the required fields from the StationVotes Table. The connection to the database is then terminated and a table is then generated using all the database information stored in the variable if all the input fields from the XHTML satisfied the conditions specified. This table was formatted by using the stylesheet from problem 1b.