Our project utilizes the framework introduced in the CS50 Finance problem set. Using PHP, HTML, and JavaScript, we created our website for Eliot housing using controller pages that process information to pass into template pages, which display the information using HTML. Our website starts with login.php, which implements login functionality by rendering login_form.php. Login.php also allows users to link to register.php, which renders the template register_form.php, in order to register if visiting the site for the first time. If the form is submitted, it redirects to index.php, a central controller which login.php also redirects to after logging in.

From index.php, the home.php page is rendered, displaying the dropdown options for group size and class year. These selections post to filter.php when the submit button is clicked, and this controller performs the necessary operations on the information in our databases in order to render filter_form.php which displays all of the rooms and combinations of rooms in Eliot that satisfy the conditions of the group. While filter_form.php has links to index.php, favorite.php, and logout.php, it can also be used to add suites to a user's "favorites" database. This was implemented with JavaScript and AJAX functionality, so when an "Add" button is clicked, that room is added to a user's "favorites" database without leaving the current page. Clicking the "Add" button posts to favorite.php using AJAX, but stays on filter.php for the user. Lastly, when the link to favorite.php is clicked from the filter form, favorite.php extracts the information on the user's favorite rooms from the phpMyAdmin database and populates favorite_form.php with this information to display the rooms.

In terms of our database, we used phpMyAdmin for our project database. We created four tables. One stored our users' names and hashed passwords. The first table with information about the rooms, called "suites," contained a complete list of rooms in Eliot with a unique id, entryway, floor, number, n, common room size, the size of each bedroom, and the average bedroom size. The table "joining_suites" has up to six fields: a unique id for the suite, n, and the unique id of each room within the suite from the table "suites." The "favorites" table contained two fields: the user's id and the suite id of one of their favorite rooms. These structures allowed for all of the elements of the filter and favorite pages to display information about the rooms. Lastly, each one of the rows displayed on the filter page was given a floor and entryway tag, which allowed for potential suite to be shown or hidden based on the checkboxes at the top of the page.

Our design took into consideration many of what we considered to be the most important features for students in Eliot to sort through potential rooms. By allowing people to see all possible combinations of rooms that fit their group size, sort based on floor or entryway, rank suites based on room size, and save a list of favorite rooms to consult on room selection day, our design aimed to provide the most desirable features for Eliotites to use in the room selection process.