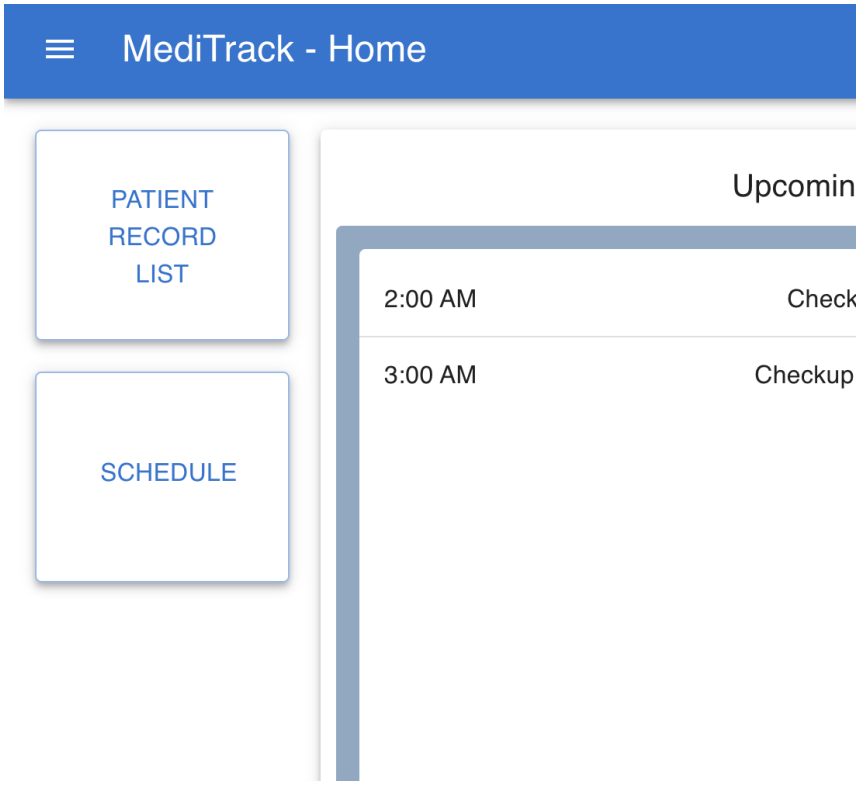
Site: https://njrchk.github.io/seng371/

User email: testing@uvic.ca

User password: Testing123

Recommendation 1: Implement a page hierarchy

Details: On the main and the other pages, there are the other two page links on the left but you cannot tell which is the home page and which page comes before which in a path.



Rationale: This will provide the user clarity about the organization of the website and also where they currently are within the site.

Suggestion on how: You can put the page navigation on the left side where the 3 lines are. When the user clicks there, the menu will expand and then we can use layering in order to show the organization. Something like this maybe:

Home

Schedule

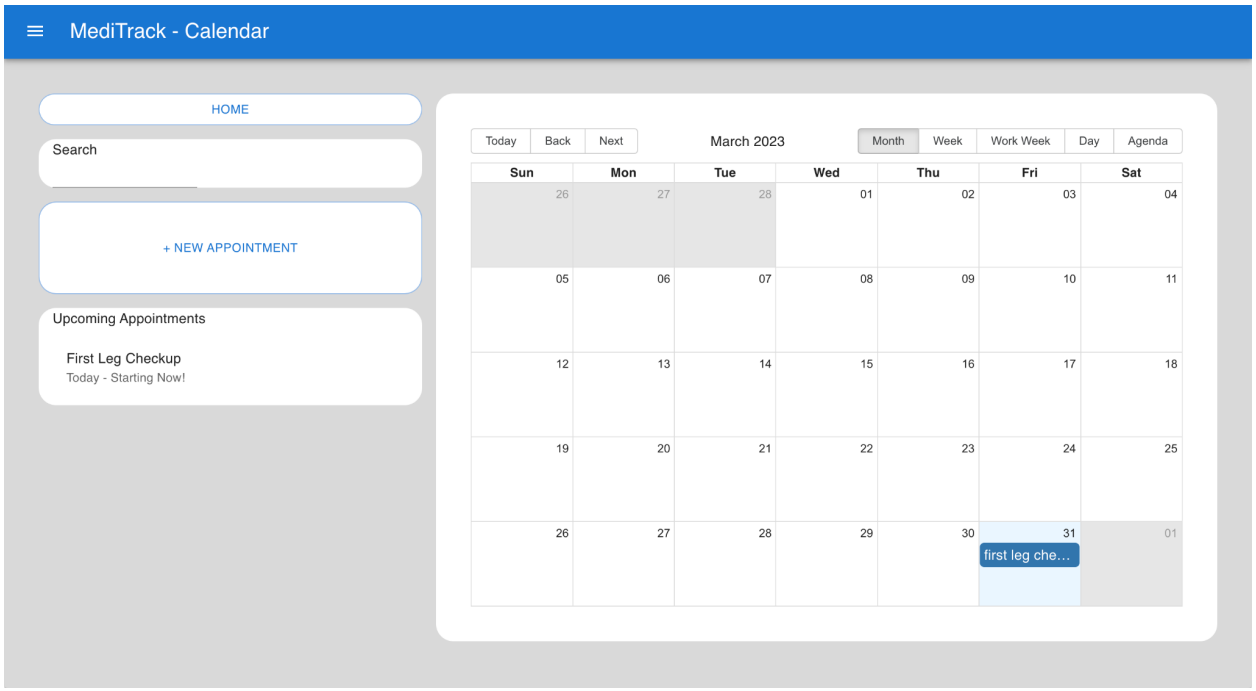
Add appointment to schedule

Patient Record List

Patient Search

Recommendation 2: Make pages navigational

Details: (This suggestion is probably redundant and will most likely get worked on anyway but just in case). When certain pages are clicked into, there is no way to return to the previous page and the user is required to reload the page.

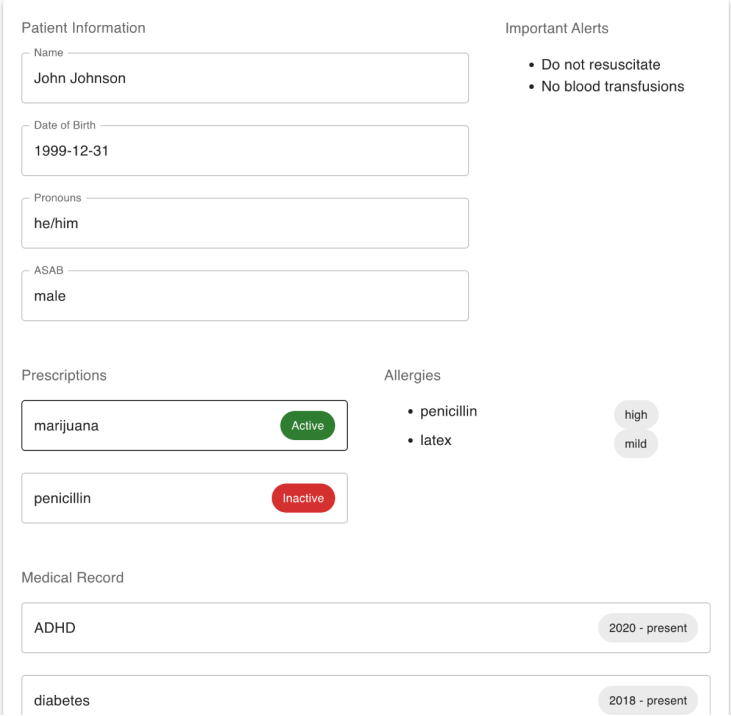


Rationale: The problem with this is that it completely logs the user out and the user has to restart the entire flow of what they were previously doing.

Suggestion on how: There are several ways to approach this, you can add back buttons, you can add links to all other existing pages the way you already have (back to the Home page), you can add paths to the URL so the user can alter their location there.

Recommendation 3: Keep page styling consistent

Details: Most pages stay consistent with styling but I noticed that the patient records page looks completely different.

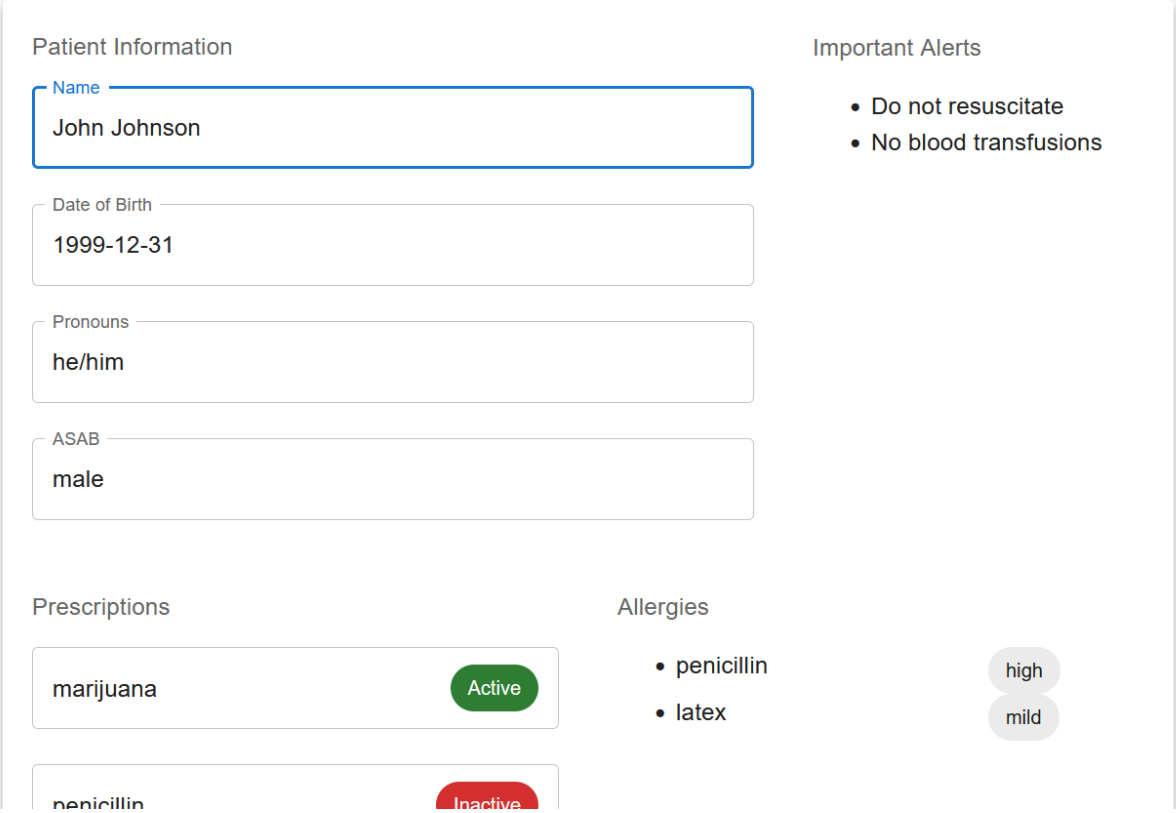


Rationale: It's okay if the page content is completely different but I think the headers and footers should stay uniform with the rest of the site just to keep up with best practices.

Suggestion on how: The easiest way to do this is to use a component library

Recommendation 4: Change the highlighting of text boxes to reflect whether they are editable or not

Details: On the patient record page, the text boxes highlight when clicked on but are not editable. In this example, the Name field is highlighted but the user is unable to edit the text in the field. Usually when text boxes aren't editable, they do not highlight when clicked on.

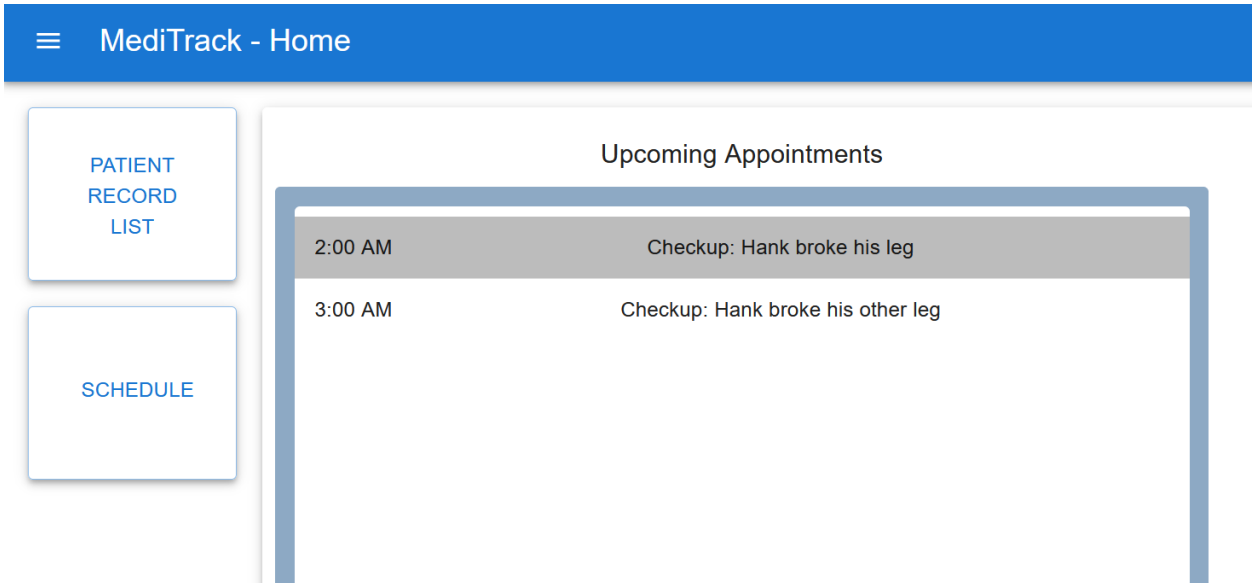


Rationale: It might be a bit confusing for users to see that they can select a text box but cannot type in it because for most software, if a user can highlight a text box then they can also edit the contents.

Suggestion on how: Change the text boxes to be unselectable or not become highlighted when selected if the currently logged in user does not have the authority to edit patient records.

Recommendation 5: Add more details to the appointment summary

Details: The appointments in the upcoming appointments section are highlighted when they are clicked on, but do not provide any further information or link to a patient's records when clicked.

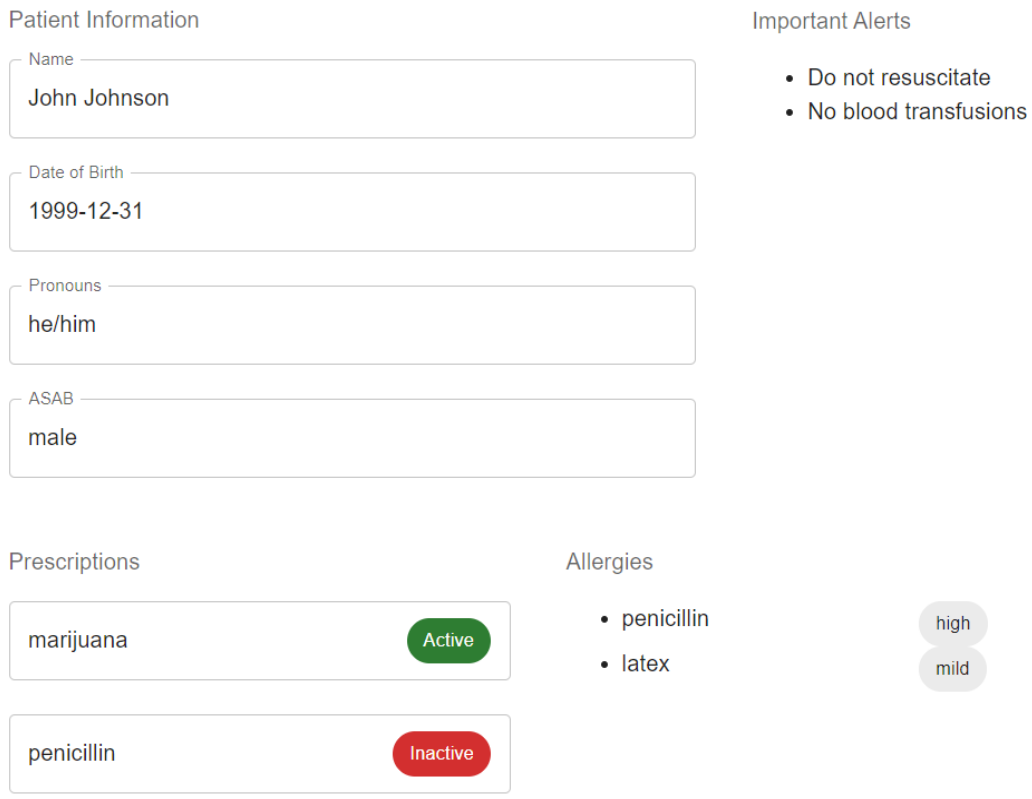


Rationale: It would be useful for healthcare professions to be able to quickly access more information about the appointment through the upcoming appointment view. This would help them save some time while using the software.

Suggestion on how: Have the appointment link to the patient's record, or expand with additional appointment details such as last appointment, etc.

Recommendation 6: Use of database

Details: It seems the entire web application does not use a database to store and retrieve the data, instead the web application data is hard coded, for example the patient information.

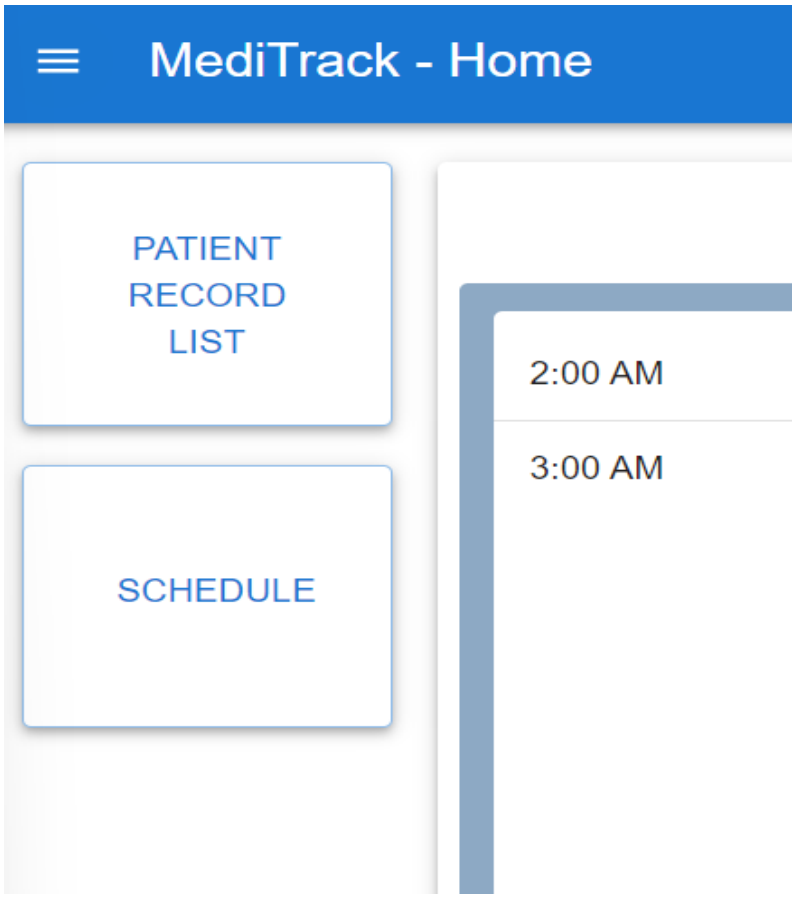


Rationale: To solve this issue, a database like Amazon Web Services can be utilized to store and retrieve the information to keep the data always up to date. For example, if a prescription for a patient is changed in the web application then having it stored inside the database can allow developers to retrieve the most updated information while having the previous prescriptions of the patient also available as a patient history inside the database. In addition, use of a database ensures security is taken care of all the time.

Suggestion on how: Connect the web application with a database and start storing the data of each necessary page to the database so it can later be retrieved as needed. To understand how to integrate a database with the web application, start with reading the documentation for the preferred database choice.

Recommendation 7: Hamburger Menu

Details: Currently, the hamburger menu is seen in all pages whether the user goes to the home page, patient record list, or the schedule section but the hamburger menu is never utilized.



Rationale: To solve this issue, the hamburger menu can have a dropdown containing home page, patient record list, and the schedule. The other solution can be to completely remove the hamburger menu (not an ideal solution).

Suggestion on how: The current home page, patient record list, and schedule can be linked to the dropdown inside the hamburger menu.