CVWO Final Submission

Jon Chua (A0135440R)

Proof of Working Application (Heroku Deployment)

Please refer to https://enigmatic-hamlet-31284.herokuapp.com/

Accomplishments

With this assignment, I have successfully built my first web application using Ruby on Rails, a framework that is new to me. This todo list has full CRUD functionality as well as the ability to add 'tags' and search for todos. Furthermore, the app has basic styling, after incorporating bootstrap. Finally, I have also learnt the basics of Git and its role in software development.

While embarking on this assignment, I have learnt the importance of self-learning. This assignment was not like any other programming assignment in NUS where all the details were given to us. Instead, we had to learn on the spot and incorporate fresh knowledge into building the todo list, which I feel accurately depicts the life of any Computer Science graduate when working in any company. In this way, we learn to become independent instead of relying on others whenever we encounter any unforeseen circumstances.

However, I have encountered some problems during this assignment. A major obstacle was having many obscure errors during Heroku deployment. Deployment to Heroku uses PostgreSQL instead of SQLite for the database and I ran into many errors while trying to deploy my app to Heroku after changing the database. Fortunately, the deployment succeeded after some troubleshooting.

Finally, there were some areas of the todo list that I felt could be improved. Firstly, I realized that the todo search for my app is case sensitive, hence the search for "todo" and "Todo" is different. I have tried implementing my own fix for this bug, however it only works on my local copy (and not the app deployed on Heroku). This may be due to the differences in database implementation (between SQLite and PostgreSQL).
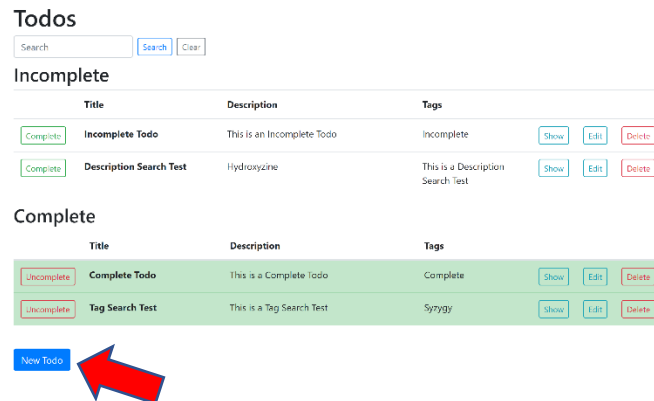
Additionally, I feel that the success screen when a todo is created or edited is redundant as the user can already see the changes on the home screen. This also applies to the 'Show' button on each todo. Furthermore, implementation of the tags was a hack as I just used a text field to store all comma-separated tags, instead of storing them separately.

All in all, this assignment was indeed an eye-opening experience for me as I experienced what it was like to be a junior software developer.
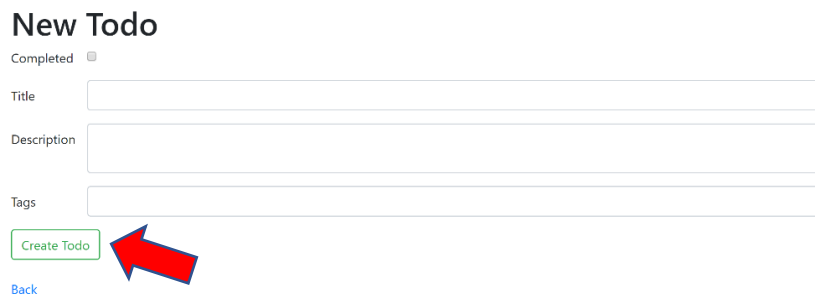
User Manual

Creating a New Todo

1. To create a new todo, click on the 'New Todo' button at the bottom of the screen as shown below.
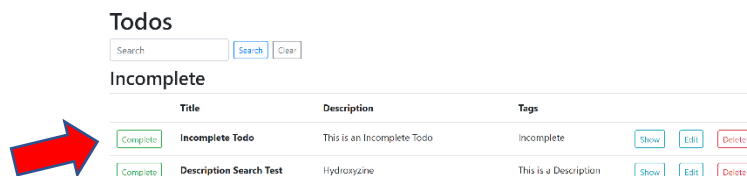

2. Enter the todo's title, description and tags and click on the 'Create Todo' button.

Marking a Todo as Complete

1. To mark a Todo as complete, click on the 'Complete' button beside the Todo that you wish to mark as complete.

Marking a Todo as Incomplete

1. To mark a Todo as incomplete, click on the 'Uncomplete' button beside the Todo that you wish to mark as incomplete.

Editing an Existing Todo

1. To edit an existing Todo, click on the 'Edit' button beside the Todo that you wish to edit.

2. Click on 'Update Todo' after you are satisfied with the changes to the Todo's contents.

Deleting an Existing Todo

1. To delete an existing Todo, click on the 'Delete' button beside the Todo that you wish to delete.

2. Click 'OK' when the confirmation prompt pops up.

Searching for Todos

1. To search for Todos, enter the keyword you wish to search for in the search bar and click on 'Search'.

2. Your search results will be shown. Click on the 'Clear' button to clear your search.