Computing for Voluntary Welfare Organisations, AY2021/22
Mid-Assignment Submission

GitHub: Main	GitHub: Frontend	GitHub: Backend	Link to working App

Introduction

I started this project with basic knowledge of HTML and CSS, and zero experience with ReactJS and Rails before. Having neither used Ruby or Go, I decided to go with the easier route and chose to use Ruby on Rails for the backend following the recommendation.

Use cases and features

Use Case 0 – User accounts and authentication

Each user to have their own account so that they are only shown their own tasks. I would also like to provide users a more convenient way to log in, by using OAuth authentication.

Users to be able to:
- Sign up for a new account by inputting email address and password
- Sign in using those credentials
- Alternatively, skip the above process by logging in via their Google or GitHub account (OAuth)

This use case has not been implemented, but is currently being worked on as of time of this submission.

Use Case 1 – Basic implementation of tasks

The most basic feature of any to-do list. Each task contains information about the task name, and the deadline.

Users to be able to:
- Create, read, update and delete (CRUD) tasks
- Mark tasks as complete

This use case is almost fully implemented. Only the marking of tasks is still not done yet.

Use Case 2 – Tagging system

A way to organise tasks. Tags are the basics required for more advanced features, such as sorting and filtering. Each tag can also be assigned a custom colour, which makes identifying related tasks easier.

Users to be able to:
- Create, read, update and delete (CRUD) tags
- Attach and detach tags to any of their tasks

This use case has been fully implemented.

Use Case 3 – Search, sort and filter

As the user makes more tasks, they need to be able to quickly sift through their tasks to the one that they are currently interested on.

Users to be able to:
- Search by task name
- Sort by deadline
- Filter by a specific tag

Other features

Features pertaining to user experience. These features are usually thought of while the app is developed. Some of the more major ones are listed here.

- Vim-based keyboard shortcuts, such as pressing '/' to search, pressing 'o' to create a new task.
- Toggling of dark mode. This is especially useful on mobile devices.
- Short in-app tutorial – Guide users on how to use the app, shown to first-time users.
- [In consideration] New tab browser extension. Replaces the new tab page with the web app. This makes it very easy for one to be always aware of what needs to be done.
- [In consideration] Telegram bot integration. Reminders can be sent to the user when one of their deadlines are approaching.

Execution plan

The current progress of my app can be found on the Issues tab of my main repo.

Do by	Features	Importance	Difficulty to implement
4 Jan	Use Case 0 – User accounts	★★★★★	★★★★★
5 Jan	Use Case 1 – (Marking of tasks)	★★★★☆	★☆☆☆☆
9 Jan	Use Case 3 – Search, sort and	★★★★☆	★★★☆☆
12 Jan	Vim-based keyboard shortcuts	★★☆☆☆	★★☆☆☆
12 Jan	Toggling of dark mode	★★★★☆	★☆☆☆☆
20 Jan	[In consideration] New tab browser extension	★★☆☆☆	★★★☆☆
20 Jan	Short in-app tutorial	★★★★☆	★★☆☆☆
22 Jan	Compatibility and responsiveness tests	★★★★★	★★★★☆

Current issues faced

I found Rails to be rather confusing to pick up. There are many configuration files that would affect the behaviour of the backend system. Moreover, I have also not worked with an SQL database before. Hence, I will dedicate more time to read the official documentations of both Rails and PostgreSQL.

Reflection

While working on my app, I've realised I had to keep switching the way of thinking, from the role of a programmer who's trying to solve the problem at hand, and to the end-user of the app. This had led me to spend dedicated time away from the code to fully ponder about how to better design the layout to make it a better experience for the user. To help me with my thought process, I created drawings of the UI layout and use them as checkpoints to work towards as I code.

A link to these drawings can be found here: Link

Also, I am now starting to appreciate how much effort is required to deliver a smooth user experience on the browser. Many subtle things have to be considered to make the user interface intuitive to anyone to use.