Name: Hannah Chia Kai Xin

Basic use case

The assignment given is to write a program that uses CRUD to create a basic to-do list, within which the tasks can be tagged so that they are easier to sort through. To-do lists are useful for many different individuals and groups. As I am new to most of these languages, before being more ambitious I will aim to implement a basic app that meets these guidelines for a single user who wants to organise different projects in their life.

As a user, opening the app will show them a table of the tasks they need to do. Buttons along the top of the page will allow them to display only those tasks tagged with a certain category. This will help to organise the to do list so it is easier for the user to see what has to be done for each category. Possible tags are for what area they pertain to(personal, project A, project B, administrative), the urgency or importance, and the estimated time frame for completion. To limit complexity, at first each category will be predetermined - if I have time, I will provide a way to create and edit categories, and filter these new categories.

Each task will have a title, a description of what has to be done, a due date, and the names of those involved. From the main page, the description will be hidden for readability. The user will be able to create, update and delete these tasks and click on the task for a view that contains the description.

After this basic functionality is in place, I will look at using status' for the tasks, so that I can mark tasks as completed and show them on a "completed tasks" page, without deleting them entirely. This would be useful to the user as they can gain a sense of achievement from looking at their completed tasks and remember what they have done already, as well as mark them as incomplete again if they realise there is more to do for that task. Marking them as complete should remove them from the main page to avoid clutter and add them to a "completed page".

Extended use case

As this assignment is to prepare for computing for voluntary welfare organisations, it might be more useful to approach this assignment as a project management tool for organisations, departments and individuals. These are stretch goals, as I am not confident in my ability to implement them within this time frame.

Usually there will be a list of projects that an agency is working on - but these would be macro level tasks, which take a long time to complete. Each project could hence be its own page, with its own description. Authentication could ensure that only those working on a project can edit the project. Having these tasks hosted on a central site could be useful to an organisation to keep everyone on the same page and have information on progress made widely available. You would be able to see what others are working on in order to keep them, and yourself, accountable.

Each project could hence have smaller tasks within it, that those working on the project can add to. It might be useful if you were able to order tasks by deadline, or by person assigned to each task.

It might also be useful for each user to have their own personal page, with their contact information and the tasks assigned to them. Managers can decide, looking at the workload the user has already taken on, if they should assign additional tasks. Individual users would be able to see all their tasks across different projects that they are responsible for. In addition, a personal page can be used to organise or order their tasks in the way they see fit to complete, giving them more freedom to personalise their own todo list without affecting the rest of the organisation.

Execution plan

Now - Jan 2: Continue readings on Ruby, Rails, REST, React, and other tools. Especially look at how to incorporate REST, as it seems to be highly recommended. If time, read resources on Typescript, which was also recommended.

January 2-9: Complete basic use cases for a single user.

The subtasks for this include:

- Finish setting up CRUD, and research how to implement tags
- Try a few different layouts with buttons, or perhaps select drop downs, in order to navigate filters.
- Decide if categories belong to their own database, or if they are a serious of variables.
- Work on a visually pleasing and clear UI

Jan 10 - 12: Stocktake, outline what to do next

Jan 12 - 25 : Buffer time + stretch goals

Problems and Suggestions

The stretch goals I would liked to get to are already outlined above. However if I have even more time, I want to explore the other areas listed in the guide especially CRON and hosting.

From trying to understand the tutorials, I think many of my problems will come from insufficient knowledge of basics and syntax. I am also worried I will end up using the wrong tool for the job, like using React to implement something that can be much more easily done with rails or a database. Hopefully taking more time to research and asking my seniors will help me to create an app that mostly follows best principles.

Also, I read that it might not be ideal to commit my whole node_modules folder, especially if where you deploy your code is different from where you write your code. However, I don't presently have the knowledge to figure out how to avoid doing so and to understand how this might affect the other user, so I've decided to commit it regardless.

I'm looking forward to trying and learning through the process!