CVWO Mid-Assignment Write-Up

Name: Sarah Tan
Matric Number: A0222470R

To-Do List Web Application

Apart from learning rails, I think that this application has got me thinking about why I really dislike making to-do lists a lot (don't worry, this is nothing related to rails or coding or programing).

Personally, I find that I spend a lot of time writing my weekly to-do list, deciding what to do from my to-do list and then recopying the stuff that I cannot finish on the to-do list to next week's to-do list. This basically takes up so much time that I spend half my life listing out what to do than actually doing anything. Hence, usually, I would just leave my brain to be the to-do list, which is not very reliable.

Thus, I wanted to pass on the Rails' philosophy DRY to my application. I want to make an application that doesn't require copying pre-existing tasks around. I also want the application to make planning out tasks for the day more productive.

Basic Functions

- Create a task
    - Click "Create Task"
    - For each task, user needs to specify the task, category, optional description, and optional due date
- Read a task
    - Click on the task to view task description
- Edit a task
    - Click on "Update Task" to update/edit a task
- Delete a task
    - Click on "Delete Task" to delete an irrelevant task
- Complete a task
    - Check the check box for the task to be completed
- Search bar
    - Search for tasks by category, due date or keywords

Other Functions (I'm Hoping to Implement)

- Account login
    - Login to account to access your to-do list
- Extend the Create Task function to have an option of including different Task types:
    - Standard Task
        - A standard task is just a normal task that has to be done by a due date and can be completed within a short period of time (ie. Homework)

    - Recurring Task
        - A task that has to be repeated every now and then (ie. doing weekly grocery shopping/ reading the newspapers)
        - The recurring task would appear in the today list automatically on the day that it should be done
    - Big Tasks with multiple sub-tasks
        - A very big task with many small tasks and spans a long time (ie. Projects)
    - Scheduled Task
        - A task that only can be done on a specific day and is one-off (ie. Bidding for module)
        - The scheduled task would appear on the today list on the day it is meant to happen
- Today list
    - Just some thoughts: The list right now is like a heap of tasks, so I'm thinking it might be helpful to have a today list to figure out what to do on the day itself.
    - Drag and drop the tasks from the heap into the today list
    - If the task is not completed, it will be added back to the heap at the end of the day.
    - If all the tasks are complete a happy dance animation is shown and the completed tasks will disappear.
- Calendar view
    - Be able to view the due dates, recurring tasks and scheduled tasks on a calendar.
- Auto addition of tasks from the luminus/ other education sites (ie Coursemology)
    - Just some thoughts: This is really wishful thinking but I think it would be nice if I don't have to type out all the assignments that are already listed on luminus.
- From the optional tasks list:
    - Cron
    - TypeScript
    - RESTful APIs
    - Hosting
    - Redux
    - Docker

Learning Journey and Problems

Honestly, I found Rails really hard to learn. I have spent the past month feeling very much confused.

Firstly, the installation really put me off. I was so confused on what I was actually installing. I was wondering for the longest time what gems are, why was I installing rbenv instead of ruby etc. One problem I had with the installation was getting Postgresql to connect to rails. I suspected it had something to do with the way I configured the username and password.

Secondly, I started learning some ruby from tutorials point (https://www.tutorialspoint.com/ruby/index.htm) and then started on rails guides (Getting Started with Rails — Ruby on Rails Guides). Even after learning some ruby syntax first, the commands and the style of writing on rails still feels rather weird and foreign to me. It felt as if different files had a different style of writing. I was also very confused why we needed so many files for an application. I felt like I was one of the blind men touching only one part of the big elephant and not seeing the bigger picture. I kind of got what the rails guides was saying, but I still did not get how to write whatever I wanted my application to do, such as functions that do not quite fall under CRUD operations.

At this point, I was feeling rather confused about what rails is and how it works so I decided to go back to the basics of what rails actually is. I found that this site from the RailsApps Project (http://railsapps.github.io/what-is-ruby-rails.html) was really useful and taught me some basic concepts that I did not have. I picked up some git here too (from the books). Git hub/ git is really daunting to me at first, there is just so much stuff that I don't understand.

At the same time, I was also learning some HTML, CSS and JavaScript from Coursera. So far, I have yet to figure out how to use react. I feel that I am rather behind on my progress on this project at this point and I still have so many wishful functions I want to implement and I am not sure if I can finish it by the deadline. I feel like I have still much to learn for Rails and React and Javascript, but I am just going to enjoy the process and clear doubts that I have and keep going.

Execution plan

Here is my rough plan for January:

- Rest of the week: Do a React tutorial and style the CRUD functions
- Week 1: Create a search bar, account login and extend the task function
- Week 2: Check out the optional tasks and try to do some
- Week 3: Calendar view, todaylist, write-up and hosting