CVWO AY19/20 – Final Assignment

Sebastian Toh Shi Jian (A0196545R)
https://github.com/sebastiantoh/todo-app

Accomplishments

1. First time building a full stack application without the need for a tutorial to handhold me from start-to-finish

Having had some prior experience dabbling with Python's Django, the MVC framework was not completely foreign to me. However, it was my first time working with Rails (and Ruby in general). Thus, I decided to start with a tutorial in order to build the backend for my application.

The process of developing using Rails was very enjoyable and newbie-friendly since Rails does most of the heavy-lifting. With that, once I felt familiar enough with Rails, I decided that I could build the backend myself without following a tutorial step-by-step. Although I kind of cheated by using the ActsAsTaggableOn gem to save myself from having to implement the tagging functionality, I managed to build a simple yet functional REST API, which my frontend uses to communicate with the backend.

With this project, it is my first full stack web application created. Previously, it was either the case that I used public REST APIs to build frontend applications, or I built a backend without a frontend.

Although my web application does not have the most features and is not the most fancy-looking, I am proud to at least have a web application that is functional and not aesthetically displeasing (at least in my opinion). For someone who only had a very superficial experience in building web applications, this whole process has been very rewarding and full of takeaways.

2. Learnt about many different aspects of web development (which I never considered): Testing, UI, UX

Testing: I learnt that there should be a systematic way to perform testing. What I did to test my application was some unstructured kind of End-to-End Testing, without a documentation of the actual steps to be performed. This often led to bugs going unnoticed. Also, I did not anticipate that there would be browser-compatibility issues. I should have incorporated testing with different browsers (specifically IE) in a much earlier stage instead of doing it only at the end.

UI, UX: I should have created a website wireframe to depict the website UI. The wireframe would make it easier to identify what should be React components in order to prevent bloated components. Although I did not do up a wireframe this time around, I had to constantly refactor my code in order to make my components leaner and to prevent code duplication.

User Manual
The application is a Single-Page Application, and all interactions will be done via the default home page.

The home page is split into 3 main sections:

1. New Task Creation
2. Filtering and Sorting Tasks
3. Tasks List

New Task Creation

This is where new tasks are created.

1. Title Input Field: a compulsory text input field and must be kept within 100 characters. No new lines are allowed.
2. Description Input Field: a compulsory text input field.
3. Tags Input Field: an optional text input field.
    a. To input tags, simply type in the name of the tag, and press "Enter".
        i. For example, if "urgent" was successfully added as a tag, you should see the following:
        

    b. The process can be repeated to add multiple tags.
    c. To delete a tag, simply click the corresponding "X" button beside the tag, or if the input field is clear of any text input, pressing "Backspace" will delete the most recently added tag (i.e. the tag furthest to the right).
    d. To clear the input field of all tags, click the "X" button located at the right-hand side of the input field.
    e. If there are already pre-existing tasks which are tagged, a dropdown menu will also appear with a list of existing tags and their corresponding frequency (i.e. number of times a particular tag is used). The options are presented in decreasing order of tag frequency.
        i. Tags can be selected from the dropdown menu by clicking on them.
    f. Note that tags are case-insensitive and will be converted to lowercase.
4. Due Date Input Field: an optional date input field.
    a. Key in the due date of your task by clicking on the input field. A prompt will appear for you to pick a date and time.
5. Once all necessary input fields are filled, clicking on the blue "ADD NEW TASK" button will create your task.
    a. Clicking on the blue "ADD NEW TASK" button without the correct input fields will result in an error. The corresponding input fields which led to the error will be highlighted with a red outline, along with a description of the error message.

Filtering and Sorting Tasks

This is where users can filter and sort their tasks. Components 1-3 below is each responsible for applying a filter (if any) on the tasks. Component 4 is responsible for sorting the tasks after the lists of tasks has gone through the series of filters.

1. Search Input Field: an optional text input field.
    a. If the input field is non-empty, a filter will be applied such that only tasks whose title or description contain the search query will appear.
    b. If the input field is empty, no filter will be applied on the tasks.
2. Tags Input Field: an optional text input field.
    a. If the input field is non-empty, a filter will be applied such that only tasks containing all the input tags will appear.
    b. If the input field is empty, no filter will be applied on the tasks
3. Hide Completed Task Switch: a toggle to hide completed tasks.
    a. When activated, a filter will be applied such that tasks which are completed will not appear.
    b. When deactivated, no filter will be applied on the tasks.
4. Custom Sorting Field
    a. Clicking on the option will render a dropdown menu, allowing users to choose how they want their tasks to be sorted.
    b. Note that sorting by "Title" is case-insensitive, and when sorting by "Due Date", tasks without any due date will always be at the bottom.

Tasks List

This is where tasks created by the user (if any) will be displayed. This is also where existing tasks can be updated.

1. By default, each task is in "View Mode".
    a. In this mode, changes made to the tags and due date will be immediately saved.
    b. Tasks which are overdue will have a red outline around the "Due Date" input field, along with a reminder.
    c. To enter "Edit Mode", click the blue "EDIT" button.
    d. To delete the task, click the red "DELETE" button.
    e. To mark the task as done, click the green "MARK AS DONE" button.
        i. Tasks which are marked as done will not have the overdue reminder even if the task is due as the task has already been completed
    f. To mark the task as not done, click the green "MARK AS NOT DONE" button.
2. Task in "Edit Mode"
    a. In this mode, any changes made will not be saved unless the blue "SAVE" button is clicked. To discard any changes made, click the red "CANCEL" button.
    b. The process of updating the task is similar to that of the "New Task Creation" section.