ENSOLVERS

Implementation exercise

1. Requirements / Intro

You need to implement a simple web application that allows to create to-do items and folders to group them. The development is divided in two phases:

- Phase 1: to-do item creation
- Phase 2: folder creation to group the to-do items

Phase 1 is **mandatory** to pass this exercise, while Phase 2 will provide **EXTRA POINTS** if done.

2. Deliverables

To pass tis exercise, in addition to the implementation, you must

- Upload a code to a public **GitHub repository** and use git properly
- Include a bash / zsh script that allows to run the app - IMPORTANT: we need to be able to start the app from a Linux/macOS environment just **running one command**. This command should setup everything that is required to run the app like, for instance, setting up a DB schema, pre-creating any config file, etc.
- Include a **README.md** file describing all the runtimes, engines, tools etc. required to run the app, with their concrete versions - for instance, MySQL 5.7, Spring Boot 2.1, React 16.10, etc.

3. Technologies

You can use whatever technology you prefer to implement the app. However, you will get **EXTRA POINTS** if you

- Structure the app as a Single Page web Application
- Use for that the following technologies
    - Angular/React for the frontend
    - Spring Boot for coding the underlying REST API

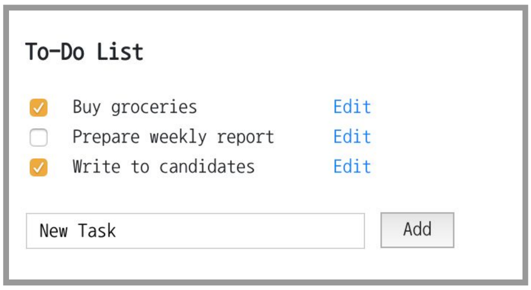
4. User Stories and mockups

Phase 1

User Stories

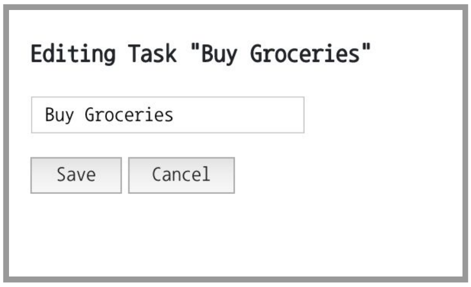
- As a user, I want to be able to create, edit and delete to-do items
- As a user, I want to mark/unmark to-do items as completed

Mockups



To-Do List

☑ Buy groceries Edit
□ Prepare weekly report Edit
☑ Write to candidates Edit



Editing Task "Buy Groceries"

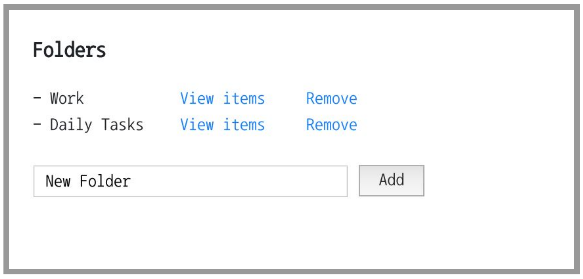
Buy Groceries

Phase 2

User stories

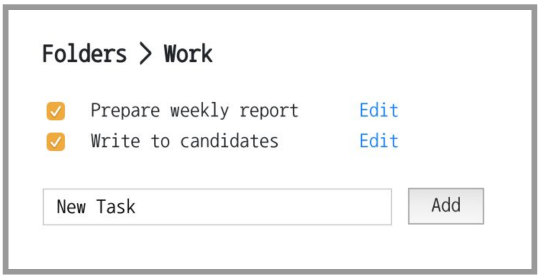
- As a user, I want to be able to create and remove new folders. NOTE: removing new folders will remove all the tasks belonging to it
- As a user, I want to be able to navigate to the item list inside a folder and manipulate the items using the same UI implemented in Phase 1

Mockups



Folders

- Work View items Remove
- Daily Tasks View items Remove



Folders > Work

☑ Prepare weekly report Edit
☑ Write to candidates Edit

5. Extra functional and non-functional requirements

These features will provide **EXTRA POINTS** to your exercise results if done

- **Login**: If you provide a login screen that allow users to log in and out from the app - you need to document the default user/password used in README.md
- **Responsive design**: If your app can be easily used in a mobile device