

# CVWO Assignment Write-up

## Task

To build a To-do manager website using Ruby on Rails, with React as frontend platform.

## Features and implementations

| Features         | Descriptions   | Implementations  |
|------------------|--|--|
| <i>Basics</i>    |  |  |
| Homepage         | Introduction of the website and a 'view lists' button that directs to '/todo_lists#index'  | Under the rails project called 'todo', generate a Homepage controller. Use React to write (under javascript folder) a 'Home' component. Link view to index.jsx under javascript/package. |
| CRUD – Todo list | The view page of todo lists will allow users to create, read, edit and delete todo lists.<br>Each todo list has a title.   | Generate a todo_lists controller with basic curd function.   |
| CRUD – Todo item | On clicking the title of each todo list shown on homepage, users will enter the url of that todo list and be able to create todo items inside the list.<br>Each todo item has a title, a description, and a checkbox.<br>The page is separated into two main components: Pending and Completed. Completed items will appear at the bottom of the page. | Generate a todo_items controller with reference to todo_lists. Edit frontend code to add links to todo list titles.  |
| Due date         | Users can select or enter due date in a todo item.   | Initialize todo_items controller with an additional column.  |
| Tagging          | Saved to-do items can be searched by their tags.   | Create a tagging model. Associate the tagging model with todo model.   |

|                     |  |   |
|---------------------|--|---|
| Empty input error   | If the title of a todo item is empty, users will not be able to save to item and an error message will pop up.   | Add some validations to models.   |
| Registration        | Users should be able to register and log in to the website, so that they can access to-do items saved before when they re-enter the website.                               | With the use of ruby gems such as devise, show data only related to user. (If needed, generate a user model?) |
| Cron                | Set up cron job to backup database.  | -   |
| Typescript          | Implement typescript in React files.   | <a href="#">Adding TypeScript   Create React App (create-react-app.dev)</a>                                   |
| Hosting             | The website should be accessible to the public.  | <a href="#">Getting Started on Heroku with Rails 6.x   Heroku Dev Center</a>                                  |
| <i>If have time</i> |  |   |
| Listing             | The view page may allow users to choose from two options of layout: 'show all' and 'show list'. 'Show all' lists out all todo items, regardless of which list they are in. |   |