

User manual:

Adding a category:

New categories can be added by typing in a category title and hitting the New Category button. Categories are unique and case insensitive.

Deleting a category:

Categories can be deleted by clicking on the delete icon at the top right corner of the category box.

Updating a category:

Category titles can be updated by clicking on the category title which opens up a text box for you to enter a new title. Hit Enter to update the title.

Adding a task:

A new task can be added under each of the categories by entering a task title, selecting an option for its priority and recurrence. Click on the Add icon to create the new task. Task titles are unique and case insensitive across all categories.

Deleting a task:

Click the delete icon along the row of the task to delete that task

Updating a task:

The task title, priority and recurrence can be updated separately. To update the task title, click on the task title which opens up a text box to enter a new title. Hit Enter to update the title. To update the priority and recurrence, click on them to open up a select box and choose an option. Clicking anywhere else closes the form and updates the task.

Sort tasks:

Tasks under a category can be sorted by opening the dropdown box and selecting an option. To sort the tasks, click on the Sort button.

Filter tasks:

Tasks titles can be filtered by entering the keyword to match and Enter button.

Note: Filter and sort do not work together, you cannot sort a list of tasks that you have filtered or filter a list of tasks that are sorted. Filter and sort only executes on the full list of tasks.

Created Rails Backend API to handle the database validations, associations, querying, and requests routes.

Used React for frontend. Made each component responsible for a single task. Created different folders to represent different logical units.

Options for recurrence and priority are retrieved from the database instead of specifying in the frontend code so that potential changes to the list of options only need to be changed once in the backend code. Refactored code for error messages to be reused across all the other components. Code for updating a task's recurrence and priority is refactored and reused to prevent duplication.