1. Use Cases
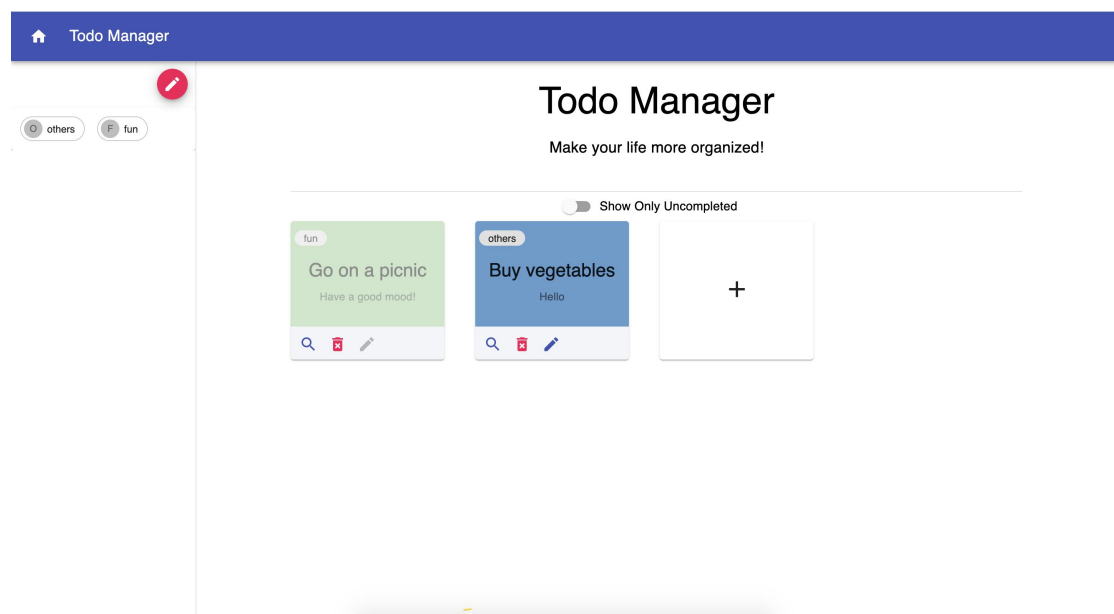
1.1. View Tasks

i). Each task is displayed in an exquisite card.

ii). A toggle is set to switch between showing all tasks and showing only those uncompleted

iii). To distinguish between completed tasks and those uncompleted, card with completed tasks are designed to be semitransparent

1.2. Sort & Filter Tasks

Filter by tags or categories is completed and others are partly completed.

1.3. Create New Tasks

In the page to create a new task, users are also able to create a new tag for the new task.

1.4. Update Tasks

Tasks cannot be modified upon being marked as completed. Users are able to update the title, description, and tag of an uncompleted task. To improve the code efficiency, the page to create a task shares the same code with the page to update a task.

1.5. View Task Info

Users are able to view more information of a task such as status, time created, time last updated, and more later ...

1.6. Complete Tasks

Users are able to mark a task as completed on the task list page (1.1) and task info page (1.5).

1.7. Delete Tasks

Users are able to delete a task on the task list page (1.1) and task info page (1.5).

2. Some Other Stuff

1.1. Execution plan

i). To finish the uncompleted features such as filter and sort tasks based on multiple keys.
ii). To come up with more interesting and creative features and implement them.

1.2. Problems Facing

i). Since I set up the project with sqlite3 but Heroku does not support sqlite3, I have to switch to postgresql in order to deploy it onto Heroku later. I wonder there are any elegant ways to do this.