Redux hooks

- create a login/register page
- use the api mocker (http://github.com/masai-school/api-mocker)
- use redux with hooks to make all the requests
- use mock server and create the following
- all api calls should be done with thunks
- create a sidebar with user profile details which you retrieve from the api mocker on the left side
- user should be able to logout from the sidebar
- create a new task page
- create an edit a task page (refer create task page, pre-fill information)
- create a summary page
- When creating a task, a task can have tags associated with it. It can be [personal, official, others], and can add multiple tags
- A task has a title, description, and a list of subtask with it, a subtask can have its own status
- A task should be given a category/task_status [Todo, In Progress, Done]
- users can toggle the status of each subtask on the main page, a user can edit the task by clicking on an edit button on the task

Screen 1 — Kanban Board

User details fetch from api mocker

All	4
Personal	1
Official	2
Others	1

LOGOUT

TODO

Task
Personal   Date
Description
☐ Sub task
☐ Sub task
☐ Sub task

IN PROGRESS

Task
Official   Date
Description
☐ Sub task
☐ Sub task
☐ Sub task

Task
Official   Date
Description
☐ Sub task
☐ Sub task
☐ Sub task

DONE

Task
Others   Date
Description
☐ Sub task
☐ Sub task
☐ Sub task

Screen 2 — Create Task

User details fetch from api mocker

All	4
Personal	1
Official	2
Others	1

LOGOUT

TITLE

DESCRIPTION

_____ ADD

☐ Sub TASK I 🗑

☐ Sub TASK II 🗑

DATE

CREATE A NEW TASK

○ Todo
○ In Progress
○ Done

Tag (multiple possible)
☐ Official
☐ Personal
☐ Others
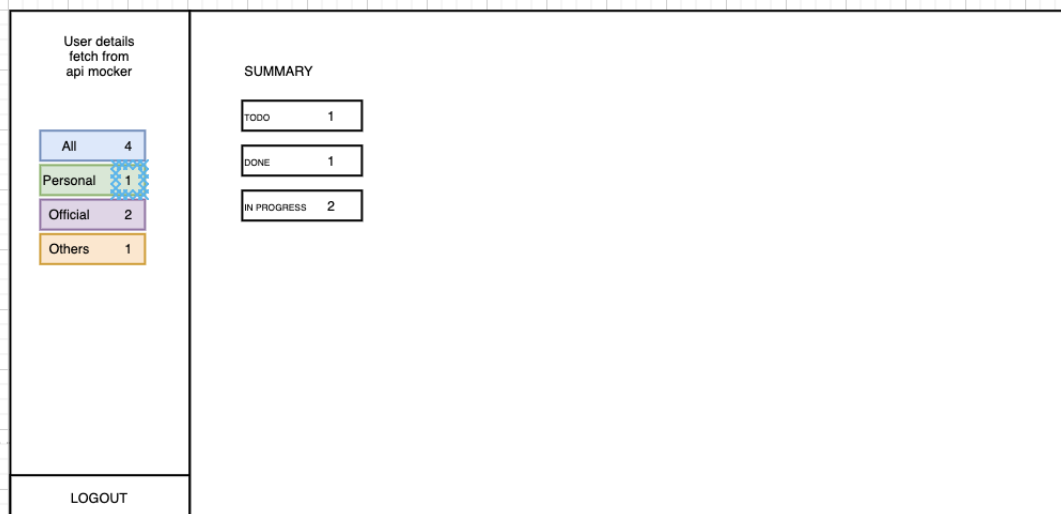
- use react-router-dom hooks to get everything working

Note:

- useDispatch, useSelector, useParams, useHistory etc to make your application

BONUS:

- use react-spring to make an animation of your choice <u>keyframes animation</u> <u>transform - mdn</u>