

Problem Statement:

You are expected to setup a Python-based server project which fulfills the functional requirements a “To-do application”.

As part of this, you will be expected to complete the following 4 tasks:

1. Writing backend API's for the To Do List application
 - a. A task will have a `title`, `description` and `due-date`.
 - b. Tasks can have related sub-tasks.
 - c. There are only 2 state applicable for task. *Pending* or *Completed*
 - d. There should be due date applicable for the task. It is fair to assume that tasks will always be sorted based on due-date (ascending).
 - e. Search (on `title`) should be available
 - f. Filtering on `due-date` should be possible using the following — Today, This week, Next week, Overdue
 - g. An alert can be configured for a task which will display a message in the UI when the alert is triggered. For an alert, assume that the user can specify the number of hours before the due date when the alert should be triggered.
2. Writing tests for the above written API's
3. All tasks which are soft deleted for more than a month should be permanently deleted from system
4. Writing minimalistic front end for the same

Pointers:

1. It would be ideal if the project is based on Django and the APIs are written using TastyPie.
2. Tests should also be in Python (Pytest preferably)
3. Front end can be anything. Can use frameworks for quick completion. A sense of clean design will certainly be much appreciated.
4. Brownie points for deploying a working version of your application on a publicly accessible URL.

Publish the work in a public repo (github or bitbucket) and share the us the link.