CMPS 350 · Web Development Fundamentals

Lab 10 · Forms and Client-side Components

Objective

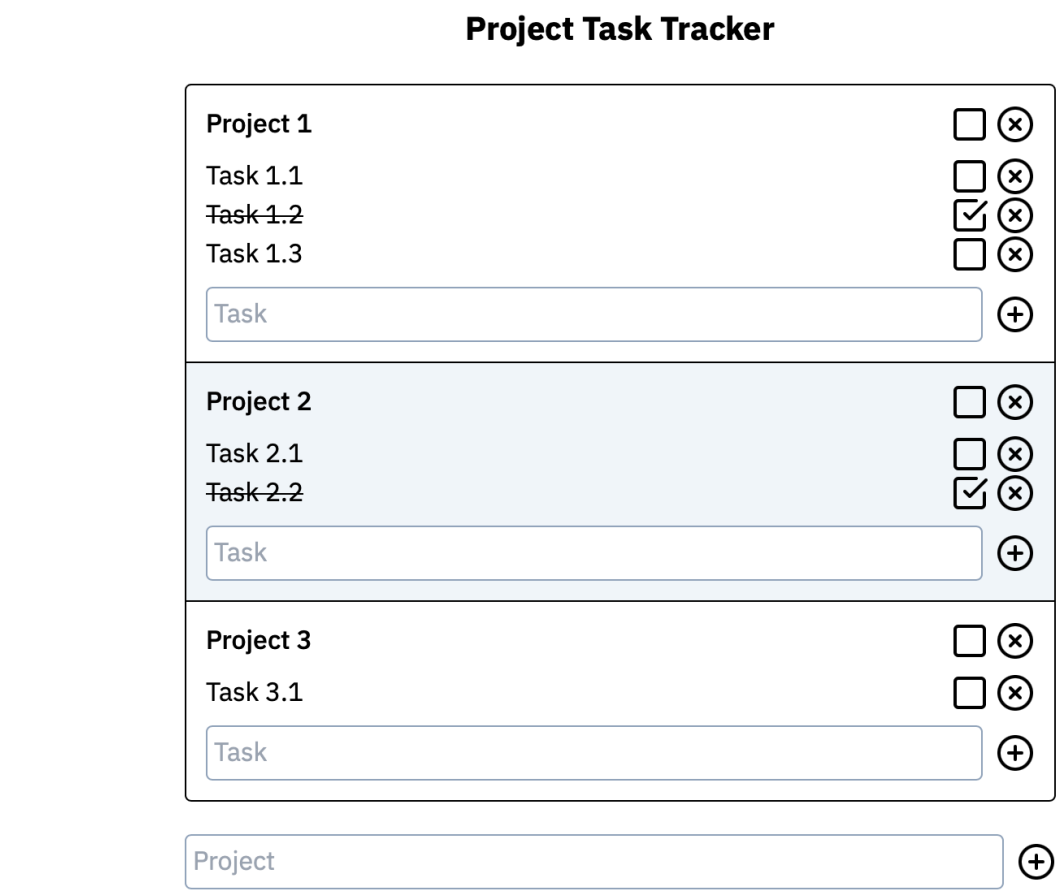
- Using hooks and handling client-side events.
- Using forms including one-to-many scenarios.
- Uploading one or multiple files.

Resources

- Learn React: https://react.dev/learn
- React docs: https://react.dev/reference/react
- Next docs: https://beta.nextjs.org/docs/getting-started
- Tailwind CSS docs: https://tailwindcss.com/doc

> We will be using Tailwind CSS to quickly style our user interfaces. It can be automatically included and configured when creating your Next applications.

1. Project Task Tracker Application



1. An interactive demo of this exercise is available at https://cmps350-project-task-tracker.vercel.app.

2. Extend the Task Tracker application to manage multiple projects with each project having multiple tasks. The base code is provided.
3. The user can add, complete, and delete tasks per project. They can also add, complete, and delete projects.
4. Projects have unique titles and tasks have unique titles within a project.
5. Handle creating new projects and tasks by pressing the `Enter` key.
6. A project should be marked as completed when all its tasks have been marked as completed.
7. Marking a project as un/completed should mark all its tasks as un/completed.

2. Photo Albums Application

1. Design and implement an application that allows the user to create albums, with required titles, and upload photos, with optional descriptions, for each album. The user can also view the list of albums and their associated photos.
2. Design, implement, and test a web API for reading, creating, updating, and deleting albums and their associated photos. An example album object is shown below. You may start by using public photo URLs from https://www.pexels.com or https://unsplash.com.

```
{
  id: "V1StGXR8_Z5jdHi6B-myT",
  title: "Summer Vacation",
  createdAt: "2023-02-25T00:00:00.000Z",
  updatedAt: "2023-03-15T00:00:00.000Z",
  photos: [
    {
      createdAt: "2023-02-18T00:00:00.000Z",
      updatedAt: “2023-02-18T00:00:00.000Z”,
      description: "Wayfarer Sunglasses",
      url: "https://images.pexels.com/photos/712395/pexels-photo-
712395.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2",
    },
    {
      createdAt: "2023-02-25T00:00:00.000Z",
      updatedAt: "2023-02-25T00:00:00.000Z",
      description: "Pair of Red-and-white Low-top Sneakers",
      url: "https://images.pexels.com/photos/914929/pexels-photo-
914929.jpeg?auto=compress&cs=tinysrgb&w=1260&h=750&dpr=2",
    },
  ],
}
```

3. Design, implement, and test a web page for viewing, creating, updating, and deleting albums and their associated photos. Display the number of photos per album and use a RAM layout for displaying the photos.
4. Design, implement, and test a web API for uploading one or many photos, as well as deleting one or many photos.

5. Extend the albums page to allow uploading and deleting photos from an album.