

# **React Internship Assignment**

**Objective:** The objective of this React Internship Assignment is to create a new React application that implements the following functionality:

## **Task 1: Application Configuration**

- Create a new React application using Typescript and Vite.
- If you are unfamiliar with Vite, please refer to the following link for more information: <https://vitejs.dev/guide/>
- Configure the application with MUI, ensuring that all components within the application (e.g. buttons, text fields, dialogs) use MUI components. If you are unfamiliar with MUI, please refer to the following link for more information: <https://mui.com/material-ui/getting-started/overview/>

## **Task 2: First Page of the Application**

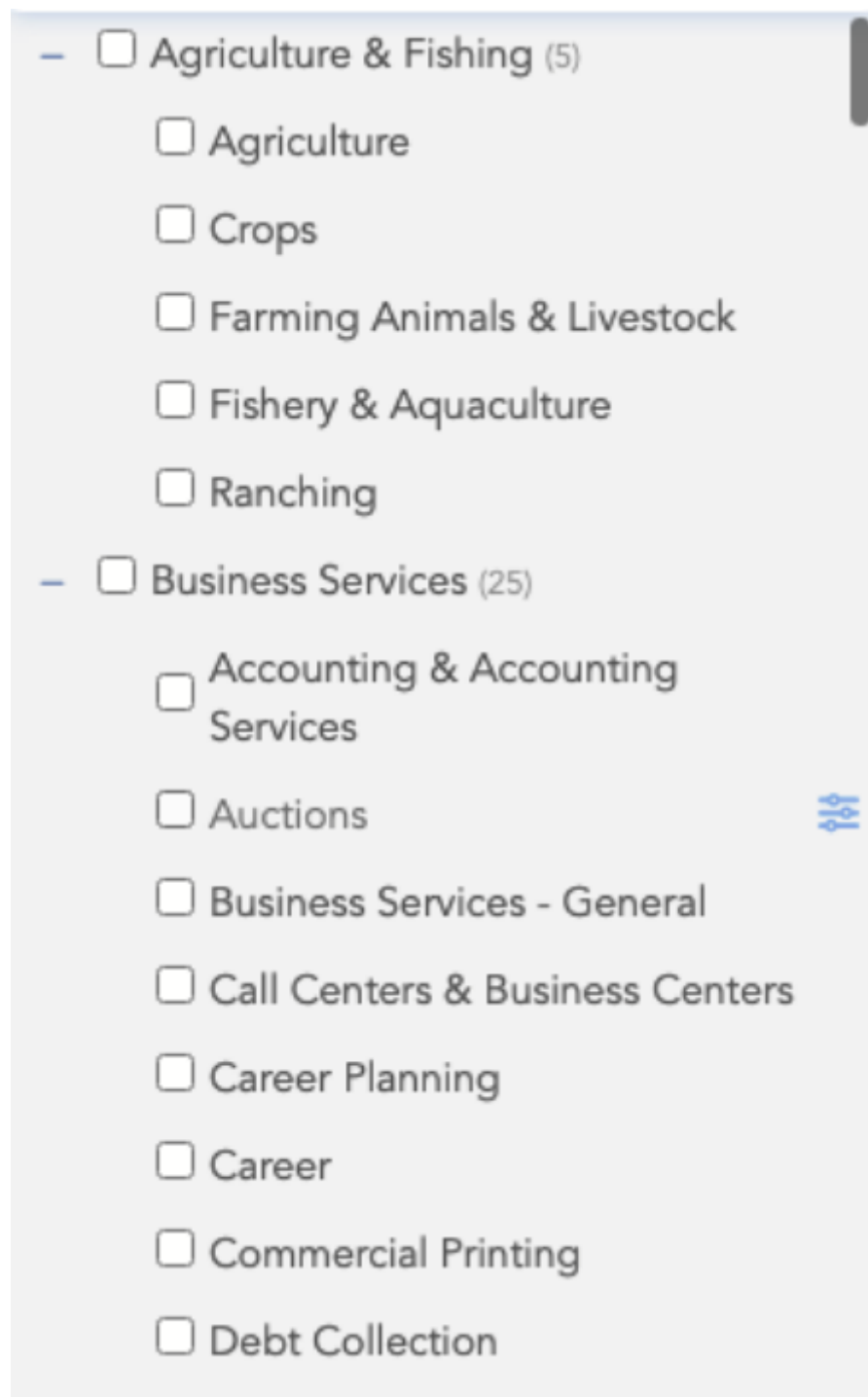
- Create a form to collect the following information from the user:
  - Name
  - Phone number
  - Email
- Upon completion, save the user details in localStorage and route the user to the second page.
- If the user attempts to access the second page without providing the necessary information, redirect them back to the first page with a message indicating that they must enter their details before accessing the page.

## **Task 3: Second Page of the Application[Component 1]**

- Fetch a list of JSON details from an API. You may use any API that returns a list of data. One example is <https://jsonplaceholder.typicode.com/posts>
- Convert the retrieved JSON data into a model/interface in Typescript. If you are unfamiliar with models or interfaces in Typescript, please refer to the following link for more information: <https://www.educative.io/blog/typescript-interfaces>
- Display the data in a table using the following package: <https://mui.com/x/react-data-grid/>

## **Task 4: Second Page of the Application[Component 2]**

- Create another component below the table from the given json(<https://docs.google.com/document/d/1uQCFny5aXfMD9v2TPy3rbE8EHw-xZVdZfj4g-2Yv6lU/edit?usp=sharing>)
- You can hardcode the given json in your application
- The component will be the list of department and each department has sub departments
- Sub departments can be expanded and collapsed by the user
- Example UI -



- 
- You don't need to be pixel perfect in terms of UI
- Users can select department or sub departments.
- If user select a department then all of the sub departments must get selected in the UI
- If user select all sub departments of a department then the parent department must get selected as well in the UI
- **YOU'RE NOT ALLOWED TO USE ANY LIBRARY TO CREATE THIS COMPONENT APART FROM MUI.**
- You can find the relevant components of MUI from [here](#).
- You can use components from the **MUI** library.

**Note:**

- The focus of this assignment is on the quality and robustness of your code, not the aesthetics of the UI.
- Clean code and proper file structure are important considerations.
- While you may use either Typescript or JavaScript for this project, Typescript is preferred.

#### **Task 4: Deployment**

- Deploy the completed application to a hosting provider. If you are unfamiliar with deploying a React application, please refer to the following link for more information:

<https://www.netlify.com/blog/2016/07/22/deploy-react-apps-in-less-than-30-seconds/>

#### **Submission:**

- You have 5 days to complete this React Internship Assignment.
- Your submission must include the Github URL for your code and the URL for the deployed application.

We look forward to your submission and the opportunity to review your work.