

SDE Internship Assignment

Contact Management - Mini Feature of a CRM



OVERVIEW

Contact Management feature helps users of the system to keep a track of important contact information of customers / clients. It lets users add, view, update, and delete contact details all in one place. This makes it easy for users to find and manage information, which is especially helpful in a business setting where keeping track of relationships is key.

For example, users can quickly find contact information, update it if anything changes, delete old contacts or add new contacts. Having all this information in one organized space saves time and keeps everything accessible, helping users build and maintain strong relationships.

Use Cases:

1. **Adding a New Contact:** Users should be able to add a contact with essential details like name, email, phone number, company and job title. This allows users to add new customers.
2. **Viewing Contacts:** A table view should list all contacts, with sorting and pagination options to make large contact lists easier to browse. This lets users efficiently locate any contact's information.
3. **Editing Contact Information:** Users can update contact details when information changes, such as a new phone number or company. Keeping data current ensures effective communication and reliable records.
4. **Deleting a Contact:** Users may need to remove outdated or duplicate entries, helping them keep their contact list clean and relevant.

IMPORTANT NOTE

This assignment will be followed by a live coding session where you will be asked to extend or modify these features on a live video call. Please complete the work independently, as we value originality and personal effort. External help or use of automated tools can easily be detected in our review of your submission and will definitely impact evaluation in a negative way.

EVALUATION CRITERIA

1. **Functionality:** Completion of all CRUD operations and the ability of each feature to perform as required.
2. **UI Consistency:** Clean and intuitive interface with well-used MUI components.
3. **Code Quality:** Clear structure, modularity, and adherence to best coding practices.
4. **Problem-Solving Approach:** How you tackled issues and designed solutions independently.

SPECIFICATIONS / REQUIREMENTS

Below are the requirements for this assignment. Please go through them with utmost attention.

Frontend (ReactJS with MUI Components)

- **Create a Contact Form:**
 - Create a form to capture new contact information with the following fields:
 - First Name
 - Last Name
 - Email
 - Phone Number
 - Company
 - Job Title
 - Use Material UI (MUI) components for all form elements, buttons, and layout.
- **Contacts Table:**
 - Display a table of all contacts using the MUI Table component.
 - Include columns for each contact attribute (First Name, Last Name, Email, Phone Number, Company, Job Title).
 - Include action buttons for editing and deleting each contact.

-
- Add pagination & sorting to the table for better usability.

Backend (NodeJS)

- **API Endpoints:**
 - **POST /contacts:** Accepts a new contact entry from the frontend and stores it in the database.
 - **GET /contacts:** Retrieves all contact entries to populate the table.
 - **PUT /contacts/:id:** Updates a specific contact's information.
 - **DELETE /contacts/:id:** Deletes a contact from the database.
- **Error Handling:**
 - Ensure validation for required fields.
 - Handle duplicate entries or any invalid data submitted.
 - Return appropriate error messages on API failures.

Database (Choice of Database)

- **Database Setup:** Use any database of your choice (e.g., PostgreSQL, MySQL, MongoDB, etc.).
- Ensure CRUD operations are correctly handled in the database.
- **Documentation:** Briefly describe your chosen database and why it fits this project.



PROJECT SUBMISSION

Code Repository: Provide a GitHub (or similar) repository link with your project code.

ReadMe File:

- Provide setup instructions to run the project, including your database schema script.
- Include a brief description of the project, any major technical decisions, and how each part of the app works.

Challenges and Solutions:

- Include a short write-up on any challenges you faced during the assignment and how you resolved them.