### Contact Info Management System for USST Project by- Abhishek S, Rishitha Reddy

The **Contact Info Management System** is a web-based application developed to streamline the process of recording and managing visitor information at **USST**. Built using Angular for the frontend, Express.js for the backend, and MongoDB as the database, the system allows administrators to create, view, update, and delete contact records efficiently.

Each contact entry includes essential personal information such as name, phone number, email, and address, along with optional **social media links** for enhanced connectivity. The system also tracks **previous visit history**, including the date and purpose of each visit, providing a comprehensive view of visitor engagement over time.

Designed with a modular architecture, the application promotes scalability and maintainability, making it suitable for both small- and large-scale organizations. The intuitive user interface ensures ease of use, while the backend API supports seamless integration with future features such as authentication, data export, or analytics.

Overall, the system improves organizational workflow by replacing manual visitor tracking with a digital, centralized solution.

## 

## 

## 

## 

## 

## 

## **Key Features**

## Add Contact

## Add name, email, phone number, and address.

## Add optional social media links.

## Add visit history (date and purpose).

## Edit Contact

## Modify any existing contact’s details including visit history and social media.

## View All Contacts

## List view with basic details like name, phone, and email.

## View Contact Details

## Full view including visit history and social links.

## Delete Contact

## Remove a contact permanently from the database.

## Search & Filter

## Filter contacts by name, date of last visit, or social media presence (if implemented).

## Validation & Error Handling

## Frontend validation and backend error responses ensure data consistency.

## **Technical Flow**

### **1. Frontend (Angular)**

* **Components**:  
  + ContactFormComponent: Used to add or edit contact details.
  + ContactListComponent: Displays the list of contacts with basic details.
  + ContactDetailComponent: Shows full information of a selected contact.
  + VisitHistoryComponent: Displays and manages previousVisits.
  + SocialMediaLinksComponent: Manages a dynamic list of social media links.
* **Services**:  
  + ContactService: Handles API requests to the Express backend (CRUD operations).
* **Workflow**:  
  + User fills a form to add/edit a contact.
  + Angular sends an HTTP request to the Express backend API.
  + Receives and displays updated data from MongoDB.

### 

### **2. Backend (Express.js + Node.js)**

* **Controllers**:  
  + Handles logic for create, read, update, delete (CRUD) operations.
  + Validates incoming data.
* **Routes**:  
  + /api/contacts/ → GET all contacts.
  + /api/contacts/:id → GET/PUT/PATCH/DELETE a single contact.
* **Middleware** :  
  + Input validation
  + Authentication (if integrated later)
* **Data Flow**:  
  + Receives requests from Angular
  + Processes using business logic
  + Communicates with MongoDB to store/retrieve/update/delete contact info

### 

### **3. Database (MongoDB)**

* MongoDB stores all contact data, including nested objects like visit history and social media links.