DESIGN DOCUMENT

1. Home Page

Welcomes the user to the web app. This page lists two options:

i. Insert a Contact
ii. Search for a Contact

To modify/delete a contact, the user must for search for a contact.

2. Insert Contact Page

- Displays a form which have all Name, Address, Phone & Date components listed as input fields.
- User has the privilege to add multiple Address, Phone & Date components, but only one set of Name components per user. Using the '+Add Address/Phone/Date' a new set of corresponding components would be appended to the form one below the other.
- User has the privilege to add all Name, Address, Phone and Date Components or can leave everything blank except FirstName just to make sense at an application level (however the database has the capacity to accept all NULL values since Contact ID is auto incremented, so each entry is unique).
- Only alphabets can be entered Name components. Only numbers can be entered in Zip Code, Area Code and Phone Number using application level JavaScript check.
- Once Submit button is clicked a success page is shown with Link to Home page. User should navigate to Home to continue further.

3. Search Contact Page

- Shows a single input field which takes a string and has a SEARCH button.
- Once search is clicked, the number of entries/hits matched with the database will be displayed. The total no of records fetched from the DB is also displayed
- A table is displayed below the search, which displays all the Name, Phone, Address, and Date components of each Contact record pulled from the database. If the user contact has multiple Address, Phone and/or Dates they are all displayed within the same row for that specific User Contact.
- Each cell in the table has a pencil symbol which the user can click to EDIT. Individual Name, Address, Phone and Date components can be edited one by one and the new value is reflected immediately on the same table.
- The last column of the table displayed on the search screen has a DELETE (X) button for each record (row) fetched from the database. If the X button is clicked that particular entry will be removed from the DB and also from the table displayed on screen.
- Once Search button is clicked, the input field will be disabled, and button will change to 'Reload and Search' which will clear the existing table and come back to initial state.

System Architecture

- The front end is created using JSP, JS, HTML and CSS.
- The application layer is created using Java and the framework used is Servlets.
- MySQL is used for the database.
- JDBC is used to connect application layer to MySQL server.
- The front end communicates with application layer using Ajax calls.
- The application layer fetches data from the database and sends back to the front end in json format.
- INSERT will first insert into Contact table and will retrieve the ID of the last inserted entry. This ID will be used to insert into other tables as a foreign key.
- SEARCH will find all occurrences of given input string in all columns and fetch the corresponding contact IDs. All data of each contact ID retrieved will be displayed on the screen.
- The front end will store the corresponding IDs of each cell (in the table) and when EDIT is initiated, the corresponding ID is used to execute an update.

Design Assumptions

- User can delete only an entire Contact record and not individual entries/components.
- User can edit individual entries but not edit entire contact record.
- User will enter valid dates in the Insert Contact form
- Each user can have any number of Phone/Address/Date types (custom).

Schema Assumptions

- Name, Address and Phone components are stored as varchar. The application will handle field validations for input fields like phone number, area code and zip code (only numeric).
- Date is stored in milliseconds in the DB and the application will convert it to date.
- All the IDs of tables are auto incremented