**Introduction:**

Contact Management System is an application which helps manage ( Search, Add, Modify, Delete) contact information (name, addresses, phone numbers, event dates).

**Architecture:**

This application uses a 3-Tier architecture – client (browser), webserver and database.

Client(Browser, HTML, CSS, Java Script, JQuery)

PHP

DataBase(MySQL)

**Database Design Info:**

|  |  |
| --- | --- |
| DATABASE: | contactlist |

Tables , Attributes and Constraints:

1. contacts:

|  |  |
| --- | --- |
| ATTRIBUTE | CONSTRAINTS |
| contact\_id | Primary key; INT ; Auto Increment ; Not NULL |
| Fname ( First Name) | VARCHAR (45); Not NULL |
| Mname (Middle Name) | VARCHAR (45); |
| Lname (Last Name) | VARCHAR (45); Not NULL |

1. address:

|  |  |
| --- | --- |
| ATTRIBUTE | CONSTRAINTS |
| address\_id | Primary key; INT ; Auto Increment ; Not NULL |
| address\_type | VARCHAR (45); NULL |
| address | VARCHAR (45); NULL |
| city | VARCHAR (45); NULL |
| state | VARCHAR (45); NULL |
| zip | VARCHAR (45); NULL |
| contact\_id | Foreign key (to contact.contact\_id);INT ;Not NULL |

1. phone:

|  |  |
| --- | --- |
| ATTRIBUTE | CONSTRAINTS |
| phone\_id | Primary key; INT ; Auto Increment ; Not NULL |
| phone\_type | VARCHAR (45); NULL |
| areacode | VARCHAR (3); NULL |
| number | VARCHAR (10); NULL |
| contact\_id | Foreign key (to contact.contact\_id);INT ;Not NULL |

1. date:

|  |  |
| --- | --- |
| ATTRIBUTE | CONSTRAINTS |
| date\_id | Primary key; INT ; Auto Increment ; Not NULL |
| date\_type | VARCHAR (45); NULL |
| date | DATE; NULL |
| contact\_id | Foreign key (to contact.contact\_id);INT ;Not NULL |

* After creating the tables, the provided CSV file is split into 4 csv files as per the table structure.
* Then, import the csv files into tables using import button in phpMyAdmin GUI.

**UI Design decisions & Assumptions:**

1. Search contacts:

* In the Main html page, a search bar is provided where the input needs to be provided by the user.
* Once the Search button is clicked, the data in search field is sent to server through POST method.
* On receiving the sent data, the data is parsed into multiple strings using space as delimiter.
* Now the query written for searching is executed and fetched results are displayed as per requirements.

1. Add New Contact:

* In the Main html page, beside the search button, another button is provided to add new contact, which redirects to another html form to collect details on new contact.
* For a new contact to be entered, First name and Last name are mandatory fields to be entered, irrespective of other fields.
* Buttons are available in form which clones and removes the fields for Adding additional fields or Deleting existing fields respectively.
* Once the form is submitted, Inline validator checks for the first and last name fields to be non-empty and also the pattern of other fields and then the form data is either sent to server through POST method or prevented from sending and incorrect fields are highlighted to be corrected.

1. Delete Contact:

* Once the Search is done in main page for a contact to be deleted, then the contact’s Full name is displayed with Modify and delete links beside the name.
* Once Delete is pressed, then the contact id of the contact is sent through

the page link to server, where the contact id is retrieved on server side and query is made to delete the contact information from all the tables.

1. Modify Contact:

* Once the Search is done in main page for a contact to be modified, then the contact’s Full name is displayed with Modify and delete links beside the name.
* Once the Modify is pressed, then contact id of the contact is sent through the page link to server.
* On Server side, queries are made to retrieve contact info from all tables in database and are displayed in form similar to Add contact form with similar features.
* On deleting any address/ phone/ date field using the delete button beside, then ajax call is made to delete that particular information from database asynchronously.
* On updating the existing or Adding a new field, UPDATE or INSERT to Database are done respectively, on pressing the save contact button.

At Webserver,

Values passed through POST method are received and Connection to Database is made in PHP using “mysqli\_connect()”.

**EER Diagram:**

