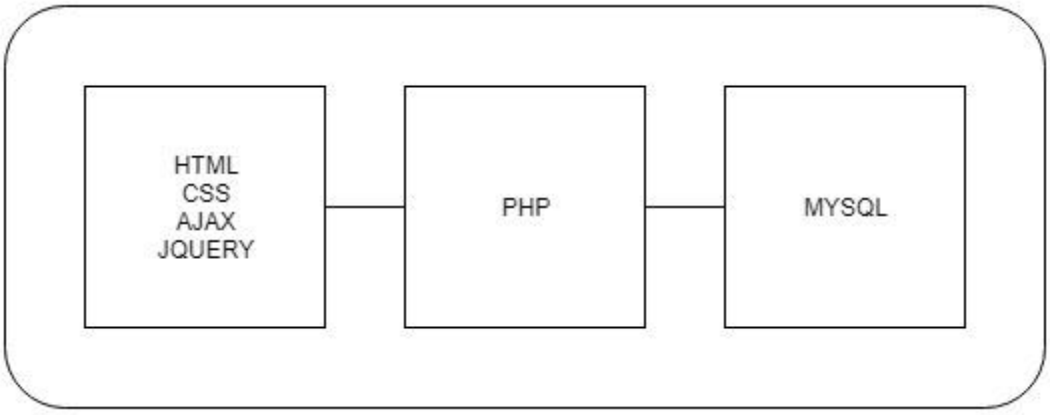
Design Document

Assumptions:

1. The application was built under the assumption that every contact is uniquely identified by a contact id.
2. This contact id that uniquely represents a contact is used in order to perform other operations on address, phone, and date tables.
3. It is assumed that a contact can give any kind of address/phone/date types (work, home, etc.).
4. The application is designed such that a contact can modify or even delete an existing address/date/phone entry.

System Architecture:

The front-end is done using HTML, CSS, AJAX, JavaScript, and jQuery. These are used in order to carry the data to the back-end. Most of the Back-end programming is done in PHP which interacts with the MySQL database to perform create, retrieve, update, and delete operations.



Design Decisions:

1. Inserting a new Contact:
    - The end user upon filling the required fields and submitting the form, expects to immediately check if he/she has been really added to the database. This can be verified by going to the search page and searching for that contact.
    - POST method in the form is used to transfer the data to the PHP script.
    - A PHP script is accessed these post variables and uses them to populate the respective tables in the database.

2. Modifying an existing contact:
    - The built system accesses all the details of the existing contact that needs to be modified.
    - This is achieved by using ajax. Ajax is used so that the data can be exchanged with a server, and update parts of the web page.
    - All this is done without refreshing the page.
    - Modifying and deleting these entries is achieved by using ajax so that user don't have to go back and forth to perform these operations.

3. Searching for an existing contact:
    - Search feature is built to look for a contact by giving any relevant information.
    - This is also built by keeping in mind that it will be used by the modifying and deleting a contact module.
    - Upon hitting search, the entire string is split into words and each of the word is searched in the database.
    - The results include all the contacts that were relevant to the search entry.
    - Next to each contact, 3 buttons are added that lets the user perform either modify or delete operation.
    - GET method is used to retrieve the contact id from the search page to perform the relevant operations.

4. Deleting a contact:
    - Deleting a contact will delete all the records from all the tables in the database that belong to the particular contact you would like to delete.
    - This is done by using a cascading effect.
    - By the end of this operation, all the records belonging to the contact would have been erased.

ER Diagram:

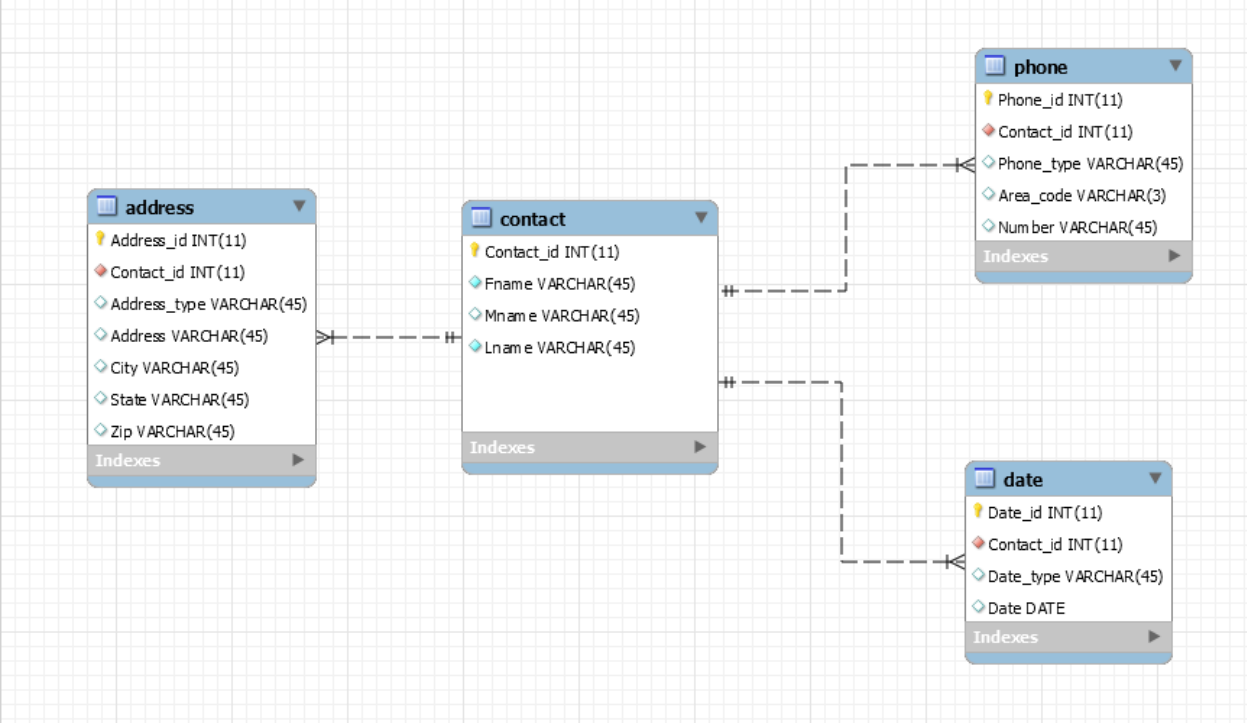

address
Address_id INT(11)
Contact_id INT(11)
Address_type VARCHAR(45)
Address VARCHAR(45)
City VARCHAR(45)
State VARCHAR(45)
Zip VARCHAR(45)
Indexes
contact
Contact_id INT(11)
Fname VARCHAR(45)
Mname VARCHAR(45)
Lname VARCHAR(45)
Indexes
phone
Phone_id INT(11)
Contact_id INT(11)
Phone_type VARCHAR(45)
Area_code VARCHAR(3)
Number VARCHAR(45)
Indexes
date
Date_id INT(11)
Contact_id INT(11)
Date_type VARCHAR(45)
Date DATE
Indexes

Table Creations:

```
CREATE TABLE `contact` (
  `Contact_id` int(11) NOT NULL auto_increment,
  `Fname` varchar(45) NOT NULL,
  `Mname` varchar(45) DEFAULT NULL,
  `Lname` varchar(45) NOT Null,
  PRIMARY KEY (`Contact_id`)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;


CREATE TABLE `Address` (
  `Address_id` int(11) NOT NULL AUTO_INCREMENT,
  `Contact_id` int(11) NOT NULL,
  `Address_type` varchar(45) DEFAULT NULL,
  `Address` varchar(45) DEFAULT NULL,
  `City` varchar(45) DEFAULT NULL,
  `State` varchar(45) Default Null,
  `Zip` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`Address_id`),
  FOREIGN KEY (Contact_id) REFERENCES contact(Contact_id)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;


CREATE TABLE `Phone` (
  `Phone_id` int(11) NOT NULL AUTO_INCREMENT,
  `Contact_id` int(11) NOT NULL,
  `Phone_type` varchar(45) DEFAULT NULL,
  `Area_code` varchar(3) DEFAULT NULL,
  `Number` varchar(45) Default Null,
  PRIMARY KEY (`Phone_id`),
  FOREIGN KEY (Contact_id) REFERENCES contact(Contact_id)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;


CREATE TABLE `Date` (
  `Date_id` int(11) NOT NULL AUTO_INCREMENT,
  `Contact_id` int(11) NOT NULL,
  `Date_type` varchar(45) DEFAULT NULL,
  `Date` date DEFAULT NULL,
  PRIMARY KEY (`Date_id`),
  FOREIGN KEY (Contact_id) REFERENCES contact(Contact_id)
) ENGINE=InnoDB AUTO_INCREMENT=1 DEFAULT CHARSET=latin1;
```