

CONTACT LIST APPLICATION

DESCRIPTION:

An online contact list application for managing contacts with basic CRUD(Add, View, Update, Delete) functionalities. It was implemented using JAVA GUI that interfaces with backend MySQL Database.

PROGRAMMING LANGUAGES AND FRAMEWORKS:

- Implemented GUI using Java Programming Language. JSwing(JFrame, JButtons, JtextField) was used to develop the frontend.
- MySQL is used at the backend to store the database of contacts in the contacts.csv file
- The Application interacts with SQL using Java Database Connection (JDBC) API in Java

FUNCTIONALITIES:

1. Graphical User Interface:

JFrame is used to design the Contact Display Window which displays the contacts. The Contact Display Window displays the following information while viewing, adding, modifying or deleting a contact:

- First Name, Middle Name and Last Name
- Address List (Type of Address, Street Address, City, State & Zipcode)
- Phone Number List (Type of number, Area Code, Number)
- Date List (Type of date, date)

A JTable Object is used to display the contacts in the JFrame.

- Search Button is used to search contact in the database
- View Contact Button is used to view the selected contact
- Add New Contact Button is used to add a new contact to the database
- Modify Contact Button is used to modify the selected contact
- Delete Contact Button is used to delete the contact from the database

JDialog is used to add new contact/modify or delete a contact. The input for JDialog Object is acquired using JLabels, JButtons and JtextField.

2. Data:

Baseline data can be uploaded using both using MySQL as well as Java. In MySQL the database is uploaded using the following command - **LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Contacts.csv'** where data is set global using command line prompt. The data can be loaded into the java class using **BufferedReader** and **FileReader**

3. JAVA Classes and its functions:

- **ContactConnectDatabase.java:** It contains methods like searchForContact(), getAllContact(), modifyContact() and addContact() which interacts the MySQL's SQL database contact_app_manager using JDBC API.
- **ContactListForm.java:** It creates an object that represents a contact required for passing to and from the SQL database via the methods that interact with SQL
- **ContactManagerTable.java:** When contacts are searched, it is searched in the JTable of the JFrame object. This class is responsible for displaying the contacts
- **ContactAppManager.java:** It creates a JFrame object and starts the application
- **ContactAdd.java:** Contact is added to the database using the JDialog object created by AddContact.java
- **LoadDatabase.java:** It is used to load the contacts.csv sample contact database.

4. Database in MySQL:

The data from csv file is loaded into a table called contacts. It is then used for inserting data into the tables created using the schema.

- **contacts table:** It contains all the fields present in the contacts.csv file. It is used to insert data into each separate tables
- **contact table:** It is used as a primary table where the primary key contact_id is autogenerated using AUTO_INCREMENT. The other fields are Fname(First Name), Mname (Middle Name) and Lname (Last Name)
- **address table:** It uses fields for address_type, address_street, city, state, zipcode where primary key is address_id and foreign key is contact_id
- **date table:** It uses fields like date_id, date_type, date_birth where primary key is date_id and foreign key is contact_id
- **phone table:** It uses fields like phone_type, area_code, pnumber(phone number) where phone_id is the primary key and foreign key is contact_id

INSERT INTO function is used to insert the respective fields from the contacts table into various tables.