Individual Project—Architecture Design Overview

Jiexuan Du(jxd170010)

1. For this project, I used multiple technologies to accomplish the requirements
2. For the front-end part, I used HTML, Bootstrap to design the application
3. For the back-end part, I used PHP and MySQL to develop the functions
4. Application is hosted locally on WAMP server, and also used the phpMyAdmin to build the database

Front End:

1. Using WAMP server to launch this project, after it is launched, web paged used the php backend to connect to the database and fetch the name of all the contact list with different phone, address and date information.
2. Each line of the combinations of record could be deleted, updated or added new information.
3. Once user click the "update" button, there will have a new page for user to update the information based on the record that user chose.
4. If user click the button of the "delete", the whole record will be disappeared and also the rest record of this contact will be also delete.
5. If user click the "add info" button, than there will be an add information page for user to add new information for the selected contact.
6. If want to insert the new contact, the new information that user input will transfer to the database using the php backend.
7. In the update page, user also can using three delete information buttons to delete the information that belongs to the select records.
8. User also can do the fuzzy search, like search the contact name, contact phone or address in the search page. Each result that satisfy the search content can be operated as update, delete or add information.

Back End & Database:

1. I used WAMP to set up a server and use phpMyAdmin to create and edit the database that I used for this project.
2. Created database and named *dbproject*. In this database, I created 4 tables which are split from the original data source, and they are named: contact, address, dates, phone according to the schema in the requirement file. Since I used csv files to import the data to the database, I edited the csv file which is provided to several csv files for importing effectively. The files are contact.csv, address.csv, date.csv and phone.csv
3. Main table is contact and other table have foreign keys to relate to it. All the update and delete are cascaded, which means delete or update would influence other tables.
4. I used pipe function in php to connect with this database and GUI. I simply join these four tables when the condition is satisfied. The condition is the *Contact_id* of all tables is equal, so that we will not have duplicated records in the database.

5. I designed the pagination function. In the *index.php* page I limit 40 records for each page. Since I used left join to show all the records that each contact has, there's a lot of records by the combination. Pagination make the records shows more obviously and easier to find the next one.
6. In the insert.php, user can add new contact and its information like phone, address or date. For the phone number, I used pattern to validate the input phone whether or not satisfy the format like XXX-XXX-XXXX; For the date input, I make the input type as date so that user can use calendar to add the date value. Also I limit the length that zip code can be input, the limit length of zip input is 5. Using PHP *$_POST()* function and HTML form to complete these operations.
7. In the search part, I used fuzzy searches make the input content can match all the components that contact, phone, address and dates may have. Also I use CONCAT() and TRIM() function in SQL sentence to make search the full name could work. Using *"First_name Last_name"* or *"First_name Middle_name Last_name"* either can get the records that satisfy the search contents.
8. Each record in index page or results of search page can be updated, added new information and delete. Using "update" button means only can change the content that the selected record have(the part that record do not have will be disabled to edit), where we get the *Contact_id, Address_id, Phone_id, Date_id* by using HTML href with parameters and *$_GET()* function of PHP. Using *empty()* function of PHP to check if all got parameters are empty. If each id is empty, then disabled the inputs of corresponding table in SQL.
9. Using "add info" button can add new information like phone, address or date of the selected contact, some like the insert operation. In this operation, I only make the *Contact_id* as the parameter that delivery to add information page, since we only need to add new information for the selected contact. Then, using the same step as in insert page—Using *$_POST()* get all the information that will be added, then using SQL sentence to insert new information for the selected contact. Notice that, in this SQL we need the WHERE sentence to make the condition correct.
10. Using "delete" button the user can delete the contact meanwhile delete all the information that belongs to this contact. In update page, there also are several button that called "delete", different from the button that in the index page or search page, those delete buttons in the update page only can delete corresponding information, like clicking "Delete Address", the address of this selected record will be remove from the table by using SQL and PHP to get the *Contact_id* and delete from address table where the record's *Contact_id* equals to the get one. If the selected record doesn't include some information like address, phone or date, the corresponding delete information buttons and the input place that can be edit both are disabled.