

Design & Architecture Document

1. Project and its Use case scenarios

The project is based on building an application 'Contact Manager'.

User scenarios are:

- **Login/Logout** : User login functionality so that only the user have access to his/her contacts. User have to enter the id and password for login into his/her account.
- **Search** : Makes easier to search the contacts using the search bar that will be installed in the app.
- **Import/Export** : Contacts can be imported/exported from the app so that user does not have to enter each and every contact on it's own.
- **Linking mail and contacts** : Let's user to find the email of the contact and redirects the user directly to the gmail page where he/she could write the email to that contact.
- **Update/delete Contacts**: It allows you to update or delete your contacts whenever user wants.
- **Contact profile photo** : User can also attach the photograph of the person whom the user is saving the contact.

Quality Attribute focussed

The Quality attribute that we have focussed is **Usability**.

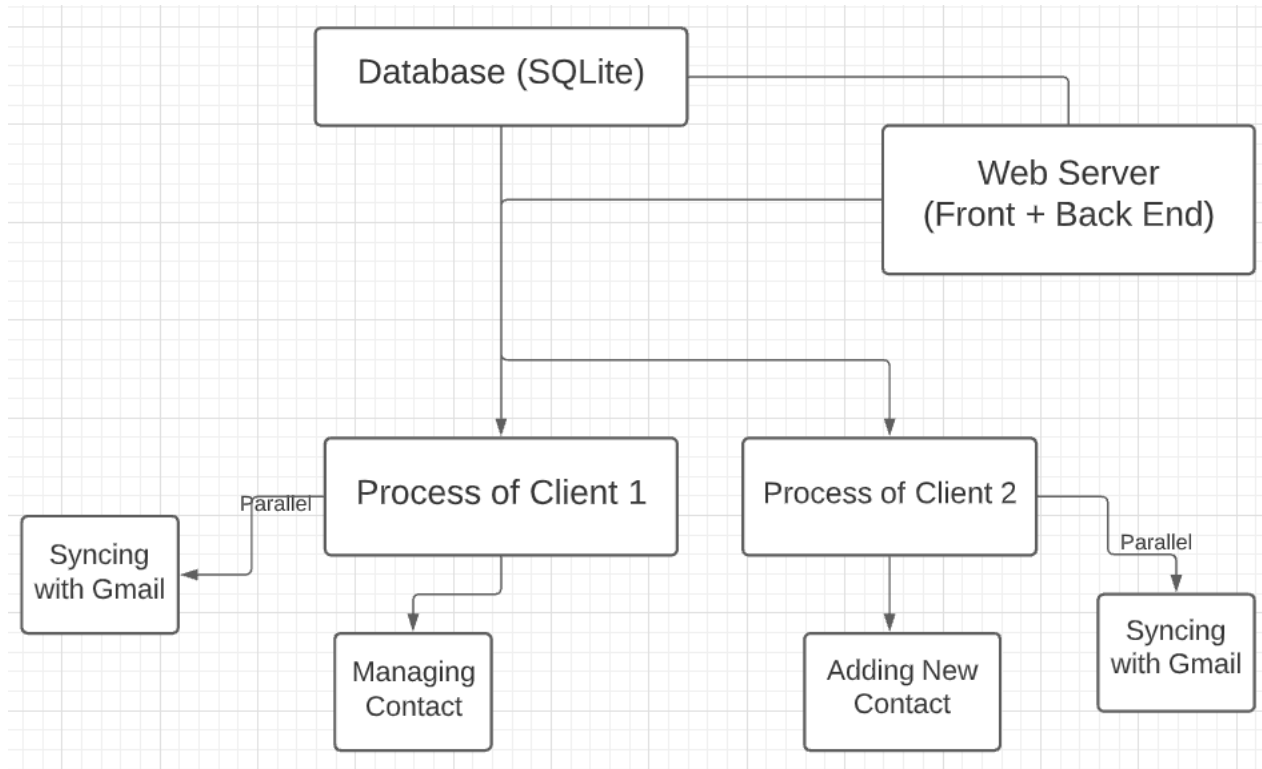
The following evaluation matrix shows how much we have concentrated our efforts on Usability.

Learnability	This webapp is designed in such a way that user could easily be able to use it's functionalities.No complexity in this app.All the contacts are displayed at the homepage itself and user easily gets the idea that where are the contacts saved,where is the profile details etc,.
Accessibility	There is no restriction on the OS.In fact it could be easily accessed through desktop,mobile,tabs and on any os like windows,android,mac os etc,.
User Interface Aesthetics	The front end is visually appealing to the users.Light colors are used all over the web app.Font size is same as default.In fact, if user uses it from mobile also then also one would find it convenient to use.Proper white spaces and indentations are kept.
Operability	It is easy to operate as all the main features which include contacts and their details,profile details could be accessed from the home page itself.
User error Protection	This attribute is most focussed upon.Our app suggests the user to make strong password and does not accepts easy passwords.Also it shows proper messages like when there are no contacts or all contacts are deleted.
Appropriateness Recognizability	As the name of the app itself "Contact Manager" describes and user immediately gets the idea that this app is about what and what all it will do.Also even if one user visits it, from the homepage itself one would be able to get all the functionalities of the app.

Explaining Design Architecture with 4+1 views

2. Physical or Deployment View

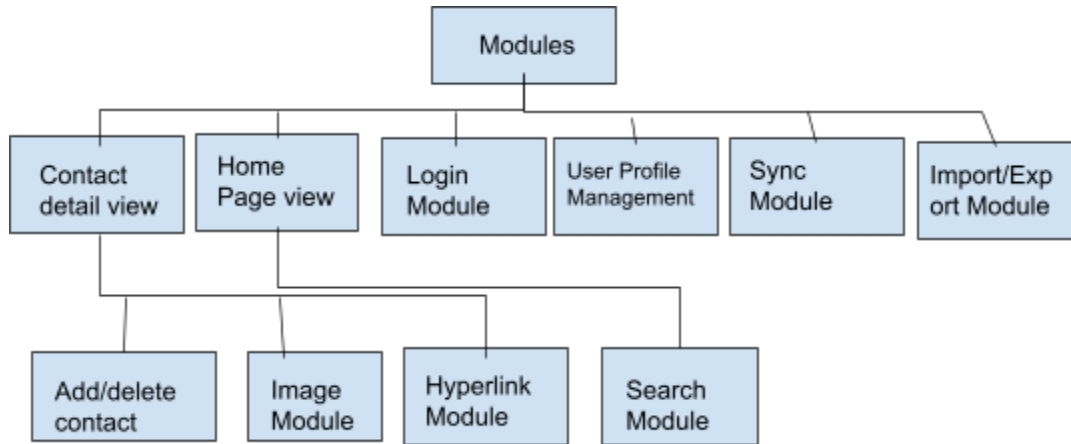
The physical view describes the components involved in the process when an application is deployed. It consists mainly of the servers where each of them plays an important role in the software deployment.



All the data of the client is saved on the database server. In the database all the user's information, user's contacts and other details are saved. The database server controls the import/export of contacts, update/delete contacts and user info. All other functionalities such as Login/Logout and linking mail and contacts are controlled by the web server using the backend. The database server is loaded on the web server. Client can then run the application on the application server and use the application.

3. Logical View

The logical view tells us about what are the different modules involved and what are their roles in the software development. It includes class diagrams and state diagrams.



We have various modules for different use case scenarios. Login Module for Login/Logout functionality, Home page view consist of search module for displaying the search functionality, User Profile Module for handling user account, Import/Export module for the contacts, Sync module for sync between different devices, and Contact details view for each contact details.

3. Development View

This view is designed for programmer's point of view. It shows how many teams are working on the software project and what are the tasks assigned to the team members and the dependencies between the teams. It is also known as implementation view.

Here we have a team of two members in our project so we will accordingly assign the task to each of us in this view.

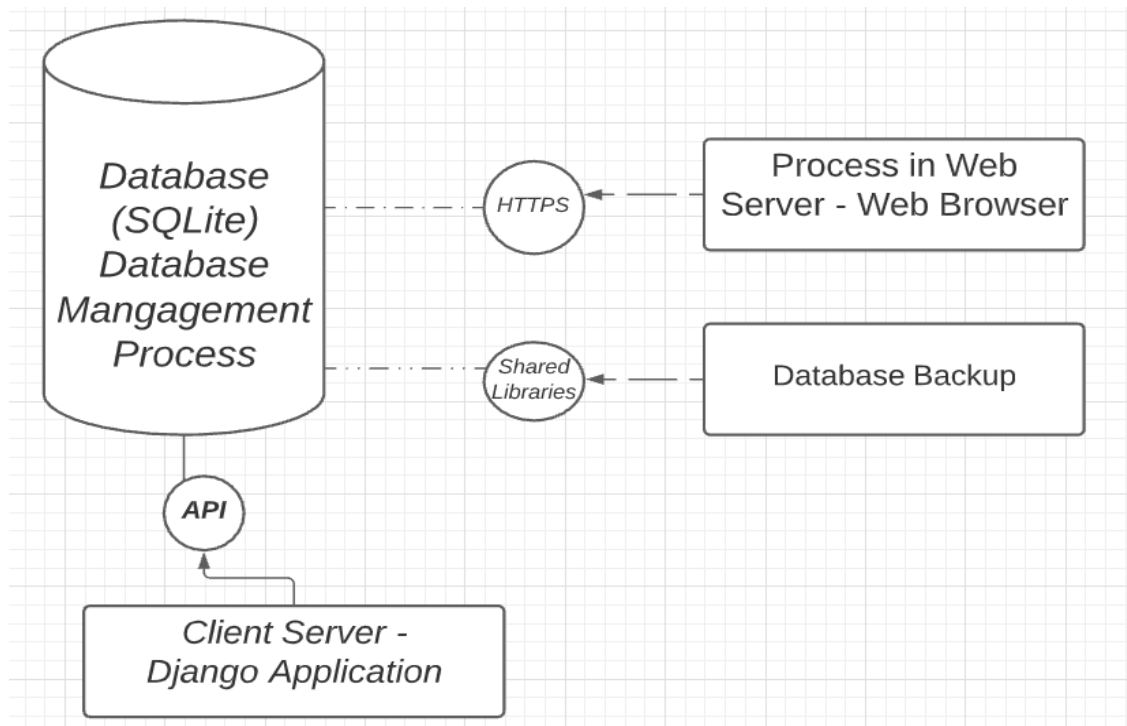
Team Member	Working on Module	Task to do
Member 1	Contact detail view, Home Page view, Login Module and mostly work on front end	Collect some contacts and their details, Home page design, code for Login functionality. Work on HTML, Js and CSS.
Member 2	Import/Export Module, Admin control, Backend, Super user	Designing code for modules and building a base for backend. Work on django framework.

All the modules cover all the use case scenarios which will be developed by the team members. Member 1 and Member 2 are assigned modules and task to

perform on each module. Some common tasks would be done collectively that include UI design, testing, maintenance and deployment.

4. Process View

This view deals with the dynamic aspects of the software and details the system processes and their communication and discusses which module runs on which physical system.



The processes that involve user scenarios run on database system. Database system involves the application server, database server, web server and database backup server.

Application runs on the application server and is connected to database server by the API. The database server has its backup stored in database backup server and uses libraries to import the backup when required. Any change in the data requires the connection of database server to the web server with the help of HTTPS and the updates data is uploaded on the web server and displayed on the web app. This is how the whole process runs on the database system.

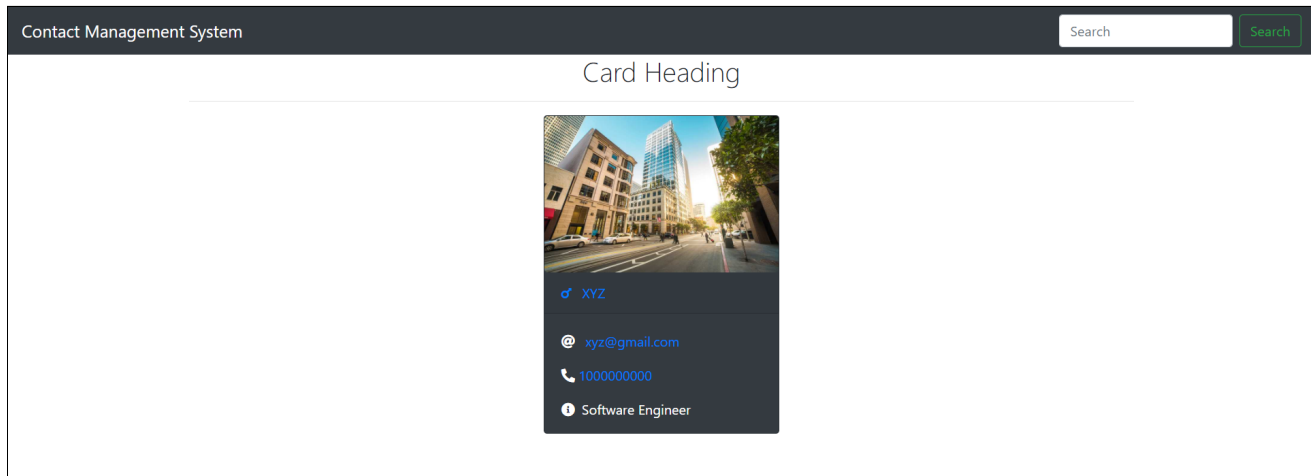
Update/delete contacts, Import/export contact, user profile, contact details are all stored in the database and controlled by the backend. Login/Logout, search, sync processes occur on the web server directly.

DESIGN WIREFRAMES

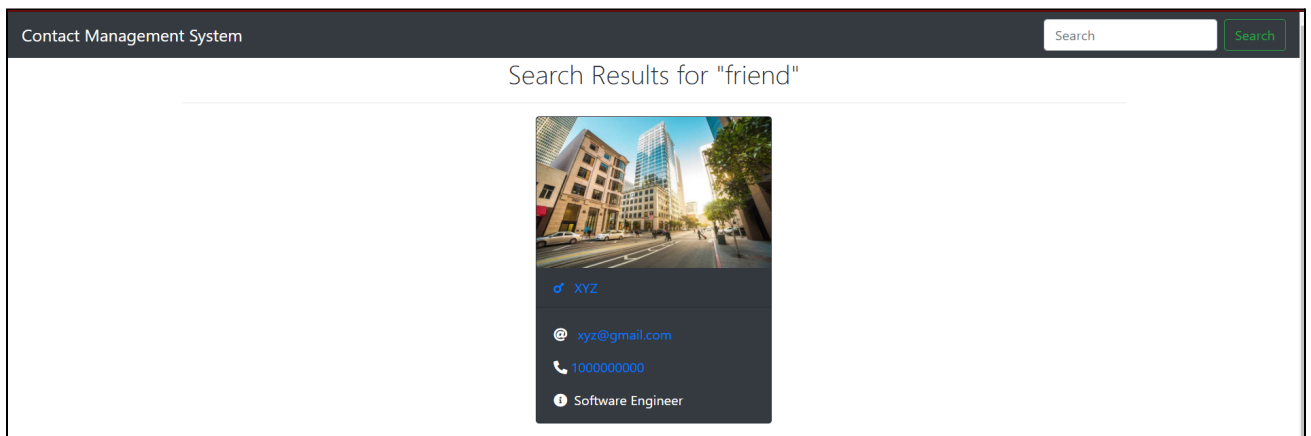
We have made wireframes by creating html pages using bootstrap for styling so that our final software project has same look and feel with the wireframes. Given below are high fidelity wireframes.

- **Homepage Wireframe**

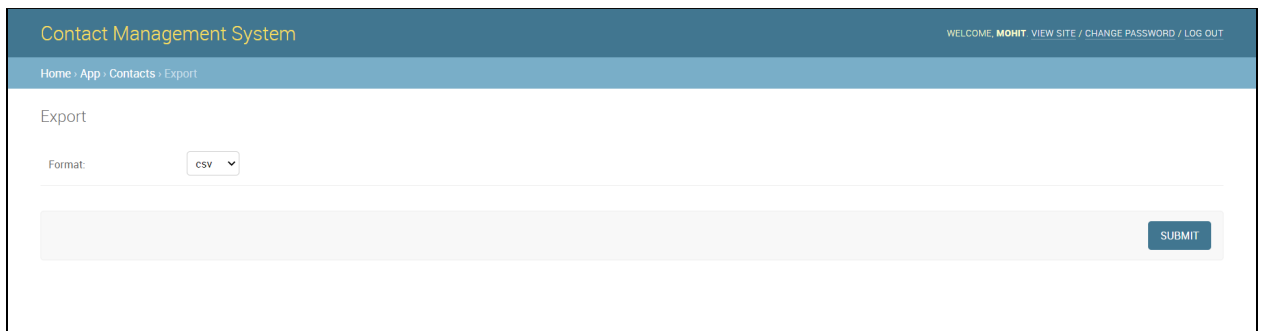
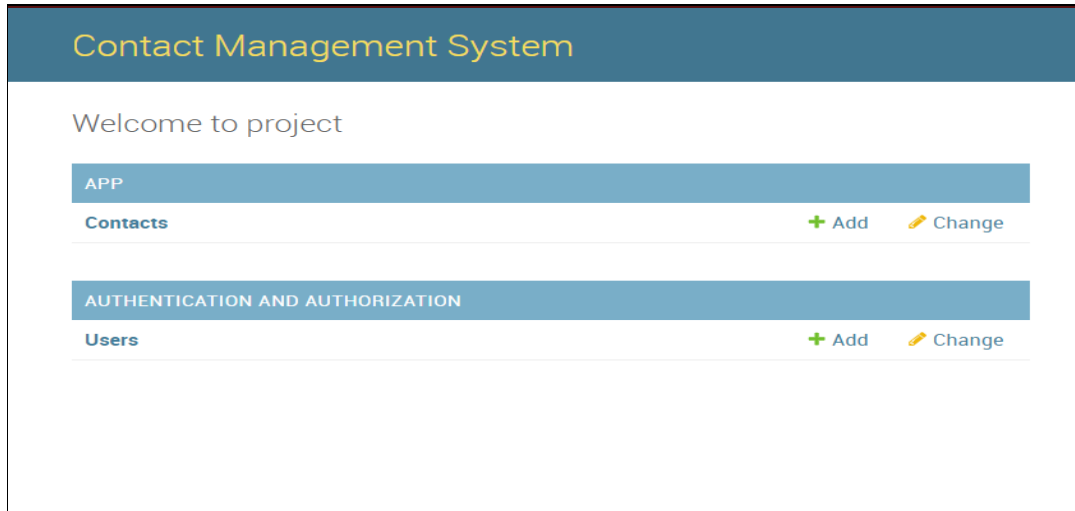
Showing details of the contact object.



- **Search Results Wireframe.**



- **Admin Page Wireframe**



- **Authentication Wireframe**

