The Project

**Patient Management Application**

Prepared By:

**Prashant Singh | Technical Trainee, at TEKsystems Global Services Pvt. Ltd.**

**About:**

This **Patient Management Application** checks if the patient is already registered and registers the patient if they’re not.

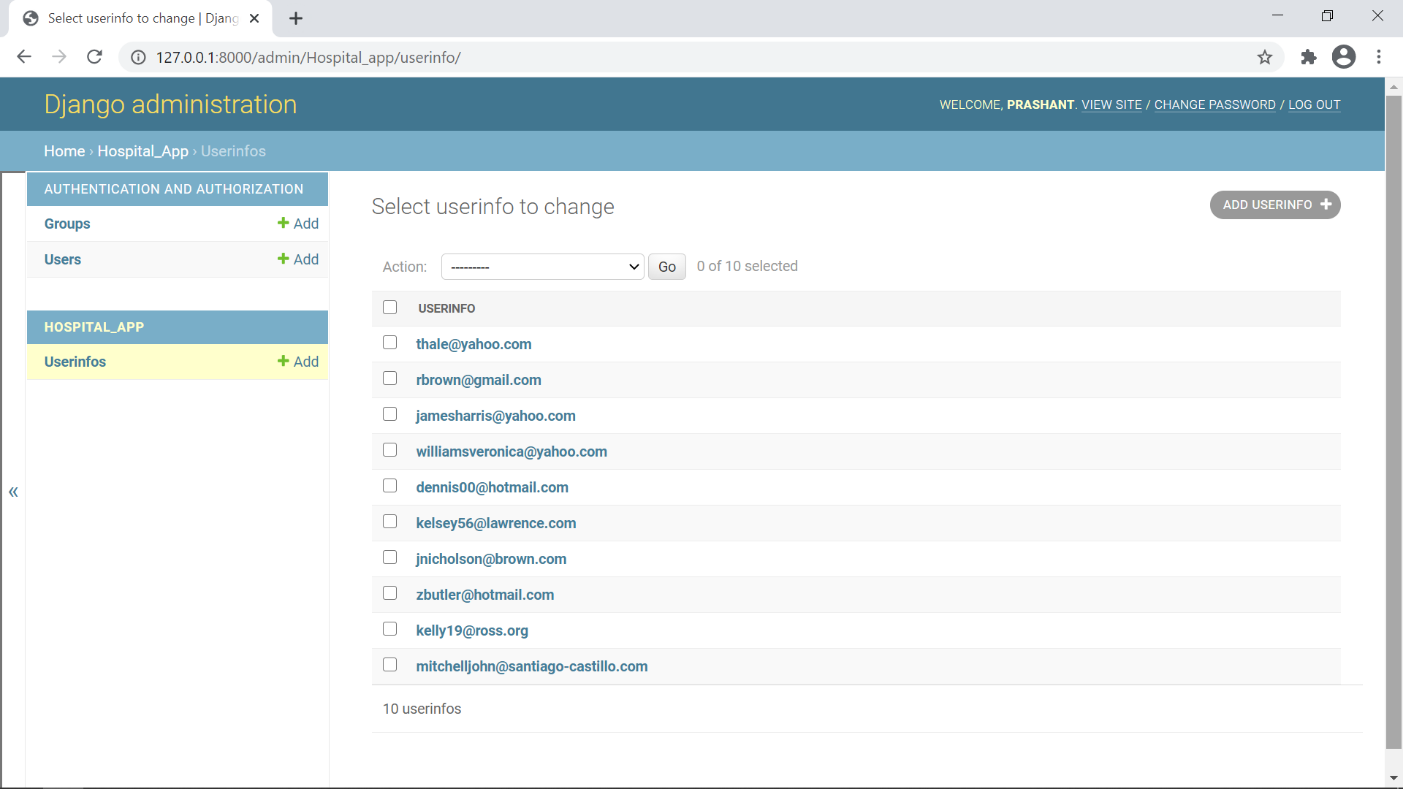
**Tools and Libraries Used:**

* Django 3.1.7
* Faker 7.0.1
* Python 3.9.2
* SQLite DB

**Overview**

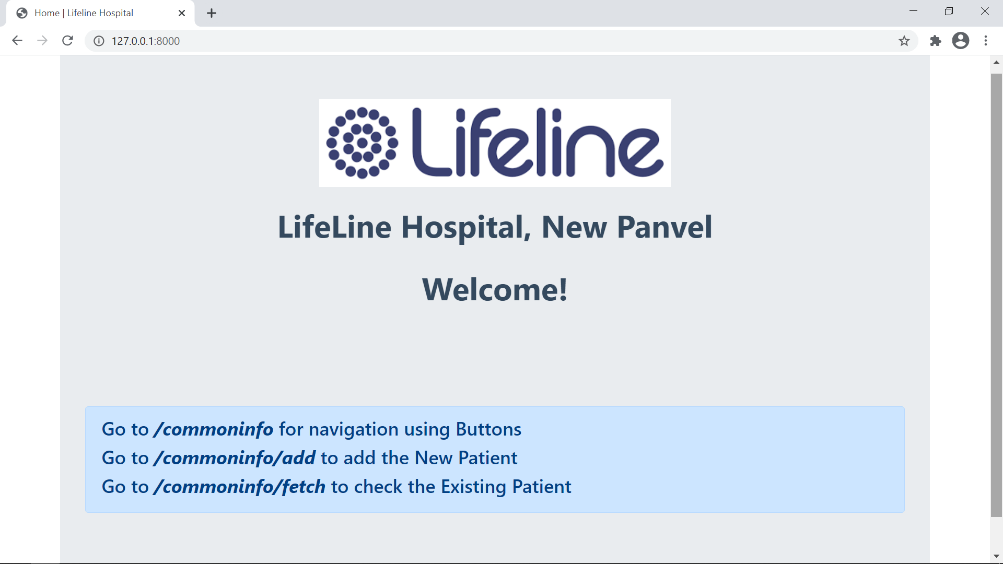
**Data Processing**

I used the Django Inbuilt SQLite database to store the patient’s information.



**Home Page**

This is the first page you will see when the " python manage.py runserver" command is issued on the terminal.



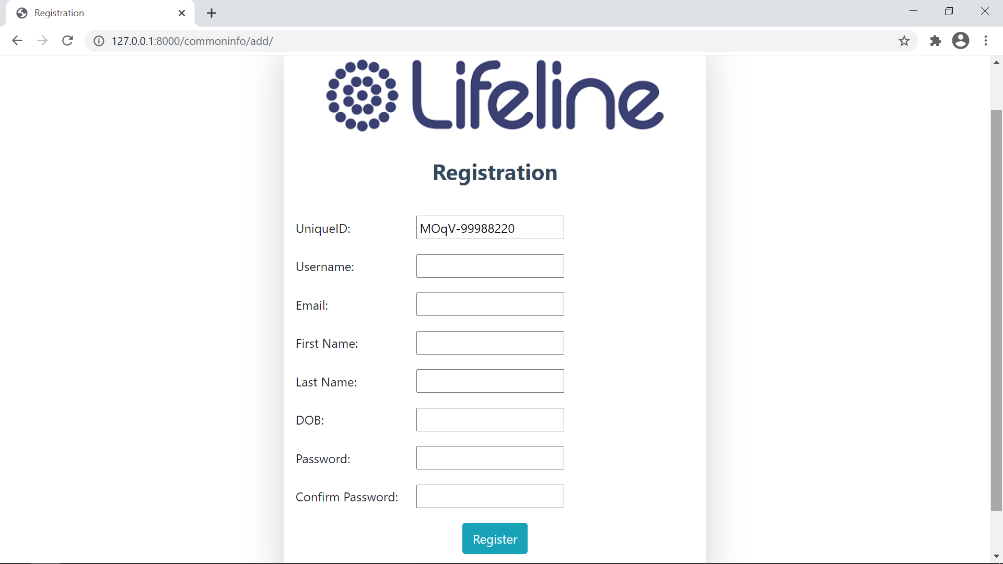
**127.0.0.1:8000/commoninfo/**

This allows user to navigate to the Registration page and Check Page using Buttons.



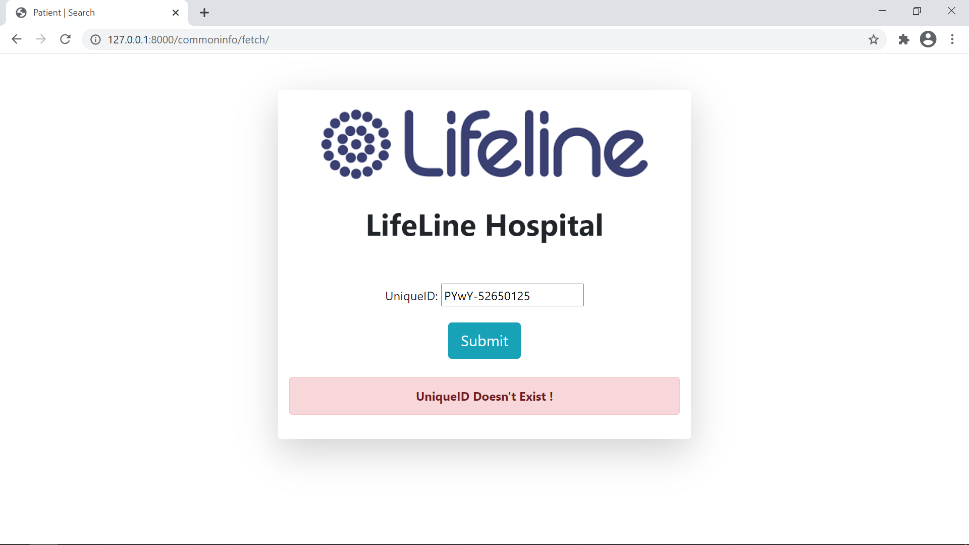
**127.0.0.1:8000/commoninfo/add/**

This is the registration form where if the patient's file is not in the database, he can register here. I have used Django **ModelForm** to make this Registration Form. All the Validations of UniqueID, Email, Password is taken care by this ModelForm.



**127.0.0.1:8000/commoninfo/fetch/**

This is the Check form where the patient's file is check in the database using UniqueID. If the file is not found in the database, it says **“Unique ID Doesn’t** **Exist”** or, if the file is found, **it displays patient information**. I have used **Django Forms** to make this Check Form.



**Conclusion**

The **Patient Management Application** is to modernize a hospital using technology. A large amount of patient data or records may also be stored. This web application is reliable enough to find large data records. During the development of this, coding standards are followed for ease of maintenance and scalability. The project has been successfully designed and is tested for accuracy and quality. During this project, I achieved all the goals and this project meets the needs of the assigned requirements.