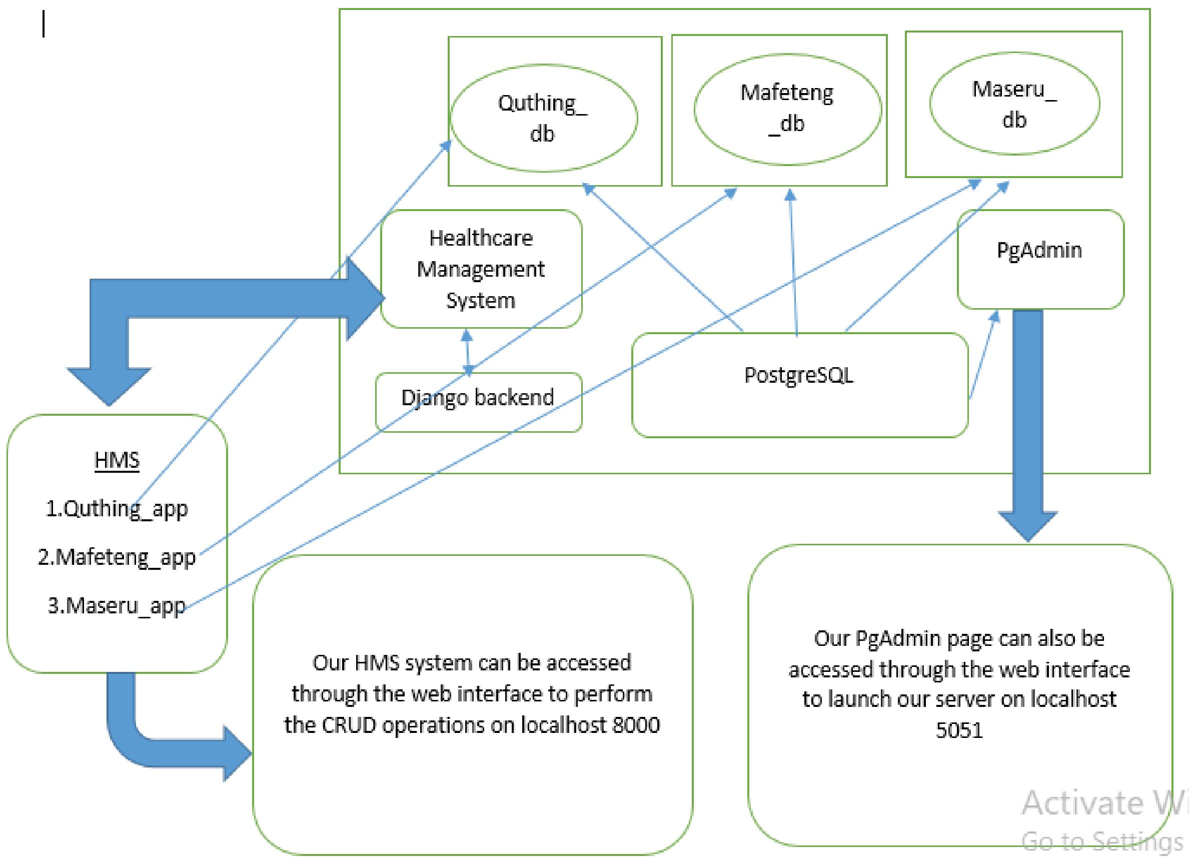
Digital Health Management System

Architectural System Design

Overview

Our digital health management system infrastructure below embodies different technologies to serve a series of tasks. We aim to create a complete guide in every aspect of those technologies to serve as a guide or a template for undertaking similar projects.

Architectural design


Quthing_ db
Mafeteng _db
Maseru_ db
Healthcare Management System
PgAdmin
Django backend
PostgreSQL
HMS
1.Quthing_app
2.Mafeteng_app
3.Maseru_app
Our HMS system can be accessed through the web interface to perform the CRUD operations on localhost 8000
Our PgAdmin page can also be accessed through the web interface to launch our server on localhost 5051
Activate Wi
Go to Settings

Component Functionalities

HMIS Application

The Django application is meant to encompass the UI, it allows the user through the admin panel to record new patients(Add/Create), Allows an admin/ Doctor to read patient info(Read), Update the user info after maybe say a diagnosis, or just their record for having visited a medical facility(Update) and finally, we give an administrator the ability to delete patient information(Delete).

The frontend is designed using our basic languages being HTML, CSS and python for integration and linkage with the server to facilitate communication with the database.

All this data from the UI, is transferred or recorded in the form of URLs, and each of those is stored in the PostgreSQL using Django's ORM.

Docker application for containerisation

We have used Docker for the containerization of our images, and or to virtualise our separate nodes.

PostgreSQL:

Has 3 containers that are carrying our Databases. These databases have been created containers for them such that they could belong to different applications that are going to be having their data distributed throughout the project.

We also have a container for our PgAdmin image, it is used to access our different databases through the web.

Healthcare Management system Application:

This is our Django application inside docker. It houses multiple apps, which of each has its own designated database of which the routers will direct to it only its designated data. We have the Quthing_app, Maseru_app, and the Mafeteng_app. All these 3 apps have databases quthing_db, naseru_db, and mafeteng_db respectively. Data that is destined for them is decided by our routers, which have our business rules regarding where to transfer each data input.

The application can be accessed through the web, through port: 8000, of which where the user will be able to input their data.

Django Application

Inside Docker we also have our Django backend for our application. The backend of the app is connected through the database using our PgAdmin server. The server has listening ports that are meant to listen to incoming HTML requests and forwards them to the designated database using queries. Inside the docker routers, the respective queries are directed again using business rules, which determine which database they are are to query.