

Patient Management Application

Course-end Project 1

Description

Many patients visit AB Hospital for a routine health checkup, accidental injuries or any other ailment treatment. Each patient has to register with the hospital if they are visiting the hospital for the first time. So, it gets very difficult for the reception boy Suresh to manage the records of every patient in a logbook manually as if the user has already registered he has to search the patient's name throughout the book containing thousands of records.

Objective: To develop an application in Django that checks if the patient is already registered and registers the patient if they're not.

Domain: Customer Support

Steps to perform:

Tasks -

1. Set up the required tools

Tools/Packages	Django 1.8 Python 2.7 SQLite	The packages are pre-installed in the environment. In case if you want to install any other package then use either apt/pip.
OS Version	Ubuntu 16.04 LTS	1 GB - 4 GB RAM and 1 - 4 vCPU.
Application port	Port 8000 is mapped port 80	Application server runs on port 8000 in the virtual machine. This port is forwarded to port 80 publicly -- which is accessible via the live preview of the IDE.

2. Do the following to execute your application

- ***python manage.py migrate***
- Click on the build button.
- You can view the Live preview of your application by appending ***/commoninfo*** in the Live preview URL i.e. "***<LivePreviewURL>/commoninfo***".
- You can visit the registration page by appending ***/add*** to the live preview URL i.e. "***<LivePreviewURL>/commoninfo/add***".
- You can visit the page to check if patient is registered or not by appending ***/fetch*** to the live preview URL i.e. "***<LivePreviewURL>/commoninfo/fetch***".

3. Use sqlite database in ***auth_user*** and create a model named ***UserInfo*** in ***models.py*** to store ***uniqueID*** and ***Date Of Birth***.

4. In **views.py** implement the logic for adding the user data in **addUserInfo** and for fetching the user data on the basis of **uniqueID** in **fetchUserInfo**.

5. Use any front-end framework to change the layout and design of the application.

- **/commoninfo**

Consists of two buttons **Register** and **Check**.



- **/commoninfo/add**

Consists of user registration form. After adding the required details and clicking on the **SUBMIT** button it should show **successfully added!!**



- **/commoninfo/fetch**

Consists of a textbox to enter UID after clicking on **SUBMIT** it should show patient's *name*, *Unique ID* and *DOB*.

