Hello Artsyy Team,

I am AVS Aditya from B. Tech. Second Year. I am so excited for the opportunity to work with the sample project given as a part of the selection process and made me to keep up the same excitement till the end of my work. I request you to go through the video sent to your mail. This video basically is on the step-wise process from an empty database till completion of project.

Hospital Management system contains 14 HTML files.

Explanation of Individual HTML File.

1. home.html – Start page of our project
2. about.html – About page
3. contact.html – Contact page
4. login.html – login page for all users (HR, Reception, Doctor and Patient)
5. registration.html is used for registering doctor and patient.
6. nregistration.html is used for registering receptionist which is visible to HR(super-user) only.
7. dashboard.html is used to show all the appointments generated by receptionist. Additional Details like total number of appointments, count of completed and active appointments…, are provided and accessible to receptionist only. Other users can track their own completed/active appointments.
8. profile.html and profile1.html are two profile pages used in this project. profile1.html will be used if user is updating his/her profile. Otherwise the system redirects to profile.html where user creates his/her own profile.
9. nurseappointment.html is a page dedicated for receptionist to create appointments
10. analytics.html is used by superuser(HR) to track total number of invoices, payment status and revenue generated.
11. invoice.html is used for displaying invoice to patient and receptionist.
12. prescription.html page is used by doctor to prescribe medicines
13. medical.html is used to view medicines prescribed to patient.
14. Showing.html is also used to show invoice when a user had consulted doctors multiple times.

There are two apps in the entire project. homeapp/urls.py contains urlpatterns of [home.html, about.html](http://home.html,about.html) and contact.html. registration/urls.py has mappings of remaining html files in the document.

All static files( css, js and images) are present in the folder static and are mapped to assets folder. Media folder contains the medical history of patient.

To start with the project first download postgresql, pgadmin4 from the following links.

<https://www.postgresql.org/download/>

<https://www.pgadmin.org/download/>

During the installation set a password as per wish and remember (here afterwards password we mean remembered password). In postgresql installation remember the username also.

After installation of both the softwares, open pgadmin4 to connect to the postgresql database. Now a popup dialog box appears. Enter the password.

After successful entry, create a server (if you are doing this for first time). Then create a database named “hospital”(as shown in the video).

Now open the project and in the vscode terminal use “pip install psycopg2” which is the POSTGRESQL database adapter. Along with that install pillow by using “pip install pillow” command. (Used for successful upload of medical history).

Now open settings.py file in the hostpital folder. In that go to Database section you must see something like this.

DATABASES = {

    'default': {

        'ENGINE': 'django.db.backends.postgresql',

        'NAME': 'hospital',

        'USER': 'postgres',

        'PASSWORD': 'kamakshi@1234',

        'HOST': 'localhost',

    }

}

There kindly change password to the remembered one.

Before opening the terminal again, kindly check if there are any migration files other than \_\_init\_\_.py in the /registration/migration folder. If any, delete.

Open your vscode terminal and use the following commands.

1. python manage.py makemigrations
2. python manage.py sqlmigrate registration 0001
3. python manage.py migrate

These commands help to convert the database models into tables in postgres. Here we are using the concept ORM (Object Relational Mapping) in Django. After executing the above three commands, check whether the django created the tables as per migration file. This is shown in the video.

Create a superuser before running the localhost server. (It is mandatory to create a Reception group).

Use the below command to create a superuser.

1. python manage.py createsuperuser

Set username, email id, password and confirm password accordingly.

Now start the server by using the command “django manage.py runserver”

Follow the video for creating users /prescription…, attached to the mail. (Similar to the video given to us)

Because of time constraints, I made the following changes in the project:

1. User Interface
2. Modification of doctor and patient details by Receptionist/HR. This may be done at admin site.
3. Filters were added in the process to ease the work of receptionist/superuser.
4. Most of the elements are covered in the attached video are discussed in detail.

Thanks for the opportunity.