**Source Code**

**url.py:**

from django.contrib import admin

from django.urls import path

from django.conf import settings

from django.conf.urls.static import static

from doctor import views as doctor

from patient import views as patient

from Medicalcare import views as medical

from hsp import views as hsp

urlpatterns = [

path('admin/', admin.site.urls),

path('index/', doctor.index, name='index'),

path('logout/',medical.logout,name='logout'),

path('admin1/', medical.adminlogin, name='admin1'),

path('adminloginentered/', medical.adminloginentered, name='adminloginentered'),

path('doctordetails/', medical.doctordetails, name='doctordetails'),

path('patientdetails/', medical.patientdetails, name='patientdetails'),

path('activatedoctor/',medical.activatedoctor,name='activatedoctor'),

path('activatepatient/',medical.activatepatient,name='activatepatient'),

path('storecsvdata/',medical.storecsvdata,name='storecsvdata'),

path('svm/',medical.svm,name='svm'),

path('decision/',medical.decision,name='decision'),

path('hsplogin/', hsp.hsplogin, name='hsplogin'),

path('hsploginentered/', hsp.hsploginentered, name='hsploginentered'),

path('uploadpatientdata/', hsp.uploadpatientdata,name='uploadpatientdata'),

path('viewpatientdata/', hsp.viewpatientdata,name='viewpatientdata'),

path('doctorlogin/',doctor.doctorlogin,name='doctorlogin'),

path('doctorregister/',doctor.doctorregister,name='doctorregister'),

path('doctorlogincheck/',doctor.doctorlogincheck,name='doctorlogincheck'),

path('doctorviewpatientdata/',doctor.doctorviewpatientdata,name='doctorviewpatientdata'),

path('addtreatment/',doctor.addtreatment,name='addtreatment'),

path('patientlogin/',patient.patientlogin,name='patientlogin'),

path('patientregister/',patient.patientregister,name='patientregister'),

path('patientlogincheck/',patient.patientlogincheck,name='patientlogincheck'),

path('treatment/',patient.treatment,name='treatment'),

path('patienttreatmentresult/',patient.patienttreatmentresult,name='patienttreatmentresult'),

]

if settings.DEBUG:

urlpatterns += static(settings.MEDIA\_URL,document\_root=settings.MEDIA\_ROOT)

**views.py:**

from collections import defaultdict

from io import TextIOWrapper

import csv

from django.shortcuts import render

from django.http import HttpResponse

from patient.models import patientModel

from patient.forms import patientForm

from doctor.models import doctorModel, storedatamodel

from sklearn.model\_selection import train\_test\_split

from sklearn.svm import SVC

from sklearn.metrics import classification\_report, confusion\_matrix

import pandas as pd

from django\_pandas.io import read\_frame

from sklearn.naive\_bayes import GaussianNB

from sklearn import metrics

from sklearn import datasets

def logout(request):

return render(request, "index.html")

def adminlogin(request):

return render(request, "admin/adminlogin.html")

def adminloginentered(request):

if request.method == 'POST':

uname=request.POST['uname']

passwd=request.POST['upasswd']

if uname =='admin' and passwd=='admin':

return render(request,"admin/adminloginentered.html")

else:

return HttpResponse("invalied credentials")

return render(request, "admin/adminloginentered.html")

def doctordetails(request):

qs=doctorModel.objects.all()

return render(request,'admin/doctordetails.html',{"object":qs})

def patientdetails(request):

qs=patientModel.objects.all()

return render(request,'admin/patientdetails.html',{"object":qs})

def activatedoctor(request):

if request.method == 'GET':

uname = request.GET.get('pid')

print(uname)

status = 'Activated'

print("pid=", uname, "status=", status)

doctorModel.objects.filter(id=uname).update(status=status)

qs = doctorModel.objects.all()

return render(request,"admin/doctordetails.html", {"object": qs})

def activatepatient(request):

if request.method == 'GET':

uname = request.GET.get('pid')

print(uname)

status = 'Activated'

print("pid=", uname, "status=", status)

patientModel.objects.filter(id=uname).update(status=status)

qs = patientModel.objects.all()

return render(request, 'admin/patientdetails.html', {"object": qs})

def storecsvdata(request):

if request.method == 'POST':

name = request.POST.get('name')

csvfile =TextIOWrapper( request.FILES['file'])

columns = defaultdict(list)

storecsvdata =csv.DictReader(csvfile)

for row1 in storecsvdata:

Pregnancies = row1["Pregnancies"]

Glucose = row1["Glucose"]

BloodPressure = row1["BloodPressure"]

SkinThickness = row1["SkinThickness"]

Insulin = row1["Insulin"]

DiabetesPedigreeFunction = row1["DiabetesPedigreeFunction"]

Age = row1["Age"]

BMI = row1["BMI"]

Outcome = row1["Outcome"]

storedatamodel.objects.create(Pregnancies=Pregnancies, Glucose=Glucose, BloodPressure=BloodPressure,

SkinThickness=SkinThickness, Insulin=Insulin,BMI=BMI,DiabetesPedigreeFunction=DiabetesPedigreeFunction,Age=Age,Outcome=Outcome)

print("Name is ",csvfile)

return HttpResponse('CSV file successful uploaded')

else:

return render(request, 'admin/storecsvdata.html', {})

**models.py:**

from django.db import models

class doctorModel(models.Model):

name = models.CharField(max\_length=50)

email = models.EmailField()

passwd = models.CharField(max\_length=40)

cwpasswd = models.CharField(max\_length=40)

mobileno = models.CharField(max\_length=50, default="", editable=True)

specialization = models.CharField(max\_length=50, default="", editable=True)

status = models.CharField(max\_length=40, default="", editable=True)

def \_\_str\_\_(self):

return self.email

class Meta:

db\_table='doctorregister'

class storedatamodel(models.Model):

Pregnancies = models.CharField(max\_length=500)

Glucose = models.CharField(max\_length=300)

BloodPressure = models.CharField(max\_length=300)

SkinThickness = models.CharField(max\_length=300)

Insulin = models.CharField(max\_length=255)

BMI = models.CharField(max\_length=255)

DiabetesPedigreeFunction = models.CharField(max\_length=255)

Age = models.CharField(max\_length=255)

Outcome = models.CharField(max\_length=255)

def \_\_str\_\_(self):

return self.Pregnancies,self.Glucose,self.BloodPressure,self.SkinThickness,self.Insulin,self.BMI,self.DiabetesPedigreeFunction,self.Age,self.Outcome

**forms.py:**

from django import forms

from doctor.models import \*

from django.core import validators

class doctorForm(forms.ModelForm):

name = forms.CharField(widget=forms.TextInput(), required=True, max\_length=100,)

passwd = forms.CharField(widget=forms.PasswordInput(), required=True, max\_length=100)

cwpasswd = forms.CharField(widget=forms.PasswordInput(), required=True, max\_length=100)

email = forms.CharField(widget=forms.TextInput(),required=True)

mobileno= forms.CharField(widget=forms.TextInput(), required=True, max\_length=10,validators=[validators.MaxLengthValidator(10),validators.MinLengthValidator(10)])

specialization= forms.CharField(widget=forms.TextInput(), required=True, max\_length=100)

status = forms.CharField(widget=forms.HiddenInput(), initial='waiting', max\_length=100)

def \_\_str\_\_(self):

return self.email

class Meta:

model=doctorModel

fields=['name','passwd','cwpasswd','email','mobileno','specialization','status']

**doctors.html:**

{% extends 'base.html' %}

{% load static %}

{% block contents %}

<div class="main-banner header-text" id="top">

<div class="Modern-Slider">

<!-- Item -->

<div class="item">

<!--<div class="img-fill">-->

<img src="{% static 'images/home.jpg' %}" alt="">

<div class="text-content">

<h3 style="color:blue"><center>Doctor REGISTRATION</center></h3>

<center> <h4><form action="/doctorregister/" method="POST">

{% csrf\_token %}

<table>

<tr><td style="color:blue">doctor name</td><td>{{form.name}}</td></tr>

<tr><td style="color:blue">Password</td><td>{{form.passwd}}</td></tr>

<tr><td style="color:blue">confirmPassword</td><td>{{form.cwpasswd}}</td></tr>

<tr><td style="color:blue">Email</td><td>{{form.email}}</td></tr>

<tr><td style="color:blue">Mobile</td><td>{{form.mobileno}}</td></tr>

<tr><td style="color:blue">specialization</td><td>{{form.specialization}}</td></tr>

<tr><td></td><td>{{form.status}}</td></tr>

</table>

<input type="submit" value="submit">

</form></h4></center>

{% if messages %}

{% for message in messages %}

<font color='darkred'> {{ message }}</font>

{% endfor %}

{% endif %}

</div>

</div>

</div>

{% endblock %}