SOURCE CODE

STEP1: IMPORTING LIBRARIES:

---

pip install numpy==1.19.2

pip install pandas==0.25.3

pip install Django==2.1.7

pip install PyMySQL==0.9.3

pip install scikit-learn==1.0.2

pip install matplotlib==3.1.1

STEP2: CREATING DATA BASE

---

create database complaint;

use complaint;

create table user_complaint(complaint_id int,

username varchar(50),

complaint_details varchar(1000),

status varchar(50),

complaint_date varchar(65));

STEP3:VIEWS.PY(CODE):

---

from django.shortcuts import render

```
from django.template import RequestContext
from django.contrib import messages
from django.http import HttpResponse
import os
import pymysql
import os
from django.core.files.storage import FileSystemStorage
from datetime import date
import numpy as np
import smtplib
import random
import speech_recognition as sr
global uname, ot
recognizer = sr.Recognizer()
def AdminLogin(request):
 if request.method == 'GET':
 return render(request, 'AdminLogin.html', {})
def UserLogin(request):
 if request.method == 'GET':
 return render(request, 'UserLogin.html', {})
def index(request):
 if request.method == 'GET':
 return render(request, 'index.html', {})
def AdminLoginAction(request):
```

```
 if request.method == 'POST':
 global uname
 username = request.POST.get('t1', False)
 password = request.POST.get('t2', False)
 if username == 'admin' and password == 'admin':
 context= {'data':'welcome '+username}
 return render(request, 'AdminScreen.html', context)
 else:
 context= {'data':'login failed'}
 return render(request, 'AdminLogin.html', context)
def sendOTP(email, otp_value):
 em = []
 em.append(email)
 with smtplib.SMTP_SSL('smtp.gmail.com', 465) as connection:
 email_address = 'kaleem202120@gmail.com'
 email_password = 'xyljzncebdxcubjq'
 connection.login(email_address, email_password)
 connection.sendmail(from_addr="kaleem202120@gmail.com",
to_addrs=em,
msg="Subject : Your OTP : "+otp_value)
def OTPAction(request):
 if request.method == 'POST':
 global uname, otp
 otp_value = request.POST.get('t1', False)
```

```
 if otp_value == otp:
context= {'data': "Welcome "+uname}
 return render(request, 'UserScreen.html', context)
 else:
 context= {'data': "Invalid OTP! Please retry"}
 return render(request, 'OTP.html', context)
 def UserLoginAction(request):
 if request.method == 'POST':
 global uname, otp
 uname = request.POST.get('t1', False)
 otp = str(random.randint(1000, 9999))
 sendOTP(uname, otp)
 context= {'data': "OTP sent to your mail"}
 return render(request, 'OTP.html', context)
def UpdateStatus(request):
 if request.method == 'GET':
 complaint_id = request.GET.get('t1', False)
 db_connection = pymysql.connect(host='127.0.0.1',port = 3306,user
= 'root', password =
'root', database = 'complaint',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "update user_complaint set status='Completed'
where
complaint_id='"+complaint_id+"'"
```

```
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 status = "Error occured during complaint update status. Please try after sometime"
 if db_cursor.rowcount == 1:
 status = "Complaint status completed"
 context= {'data': status}
 return render(request, 'AdminScreen.html', context)
def ViewComplaints(request):
 if request.method == 'GET':
global uname output = ''output+='<table border=1 align=center width=100%><tr><th><font
size="" color="black">Complaint ID</th><th><font size="" color="black">Username</th>'
 output+='<th><font size="" color="black">Complaint Details</th>'
 output+='<th><font size="" color="black">Complaint Status</th>'
 output+='<th><font size="" color="black">Complaint Date</th>'
 output+='<th><font size="" color="black">Update Complaint Status</th></tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root',
database = 'complaint',charset='utf8')
 with con:
 cur = con.cursor()
```

```
 cur.execute("select * FROM user_complaint where status =
'Pending'")
 rows = cur.fetchall()
 for row in rows:
 output+='<td><font size="" color="black">'+str(row[0])+'</td>'
 output+='<td><font size="" color="black">'+row[1]+'</td>'
 output+='<td><font size="" color="black">'+row[2]+'</td>'
 output+='<td><font size="" color="black">'+row[3]+'</td>'
 output+='<td><font size="" color="black">'+row[4]+'</td>'
 output+='<td><a href=\'UpdateStatus?t1='+str(row[0])+'\'><font
size=3
color=black>Click Here to Update Status</font></a></td></tr>'
 output+= "</table></br></br>"
 context= {'data':output}
 return render(request, 'AdminScreen.html', context)
def DeleteComplaint(request):
 if request.method == 'GET':
 complaint_id = request.GET.get('t1', False)
 db_connection = pymysql.connect(host='127.0.0.1',port = 3306,user
= 'root', password =
'root',
database = 'complaint',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "delete from user_complaint where
complaint_id='"+complaint_id+"'"
```

```
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 status = "Error occured during complaint deletion. Please try after sometime
if db_cursor.rowcount == 1:
 status = "Your Complaint Deleted with Complaint ID = "+str(complaint_id)
 context= {'data': status}
 return render(request, 'UserScreen.html', context)
def TrackComplaint(request):
 if request.method == 'GET':
 global uname
 output = ''
 output+='<table border=1 align=center width=100%><tr><th><font size=""
color="black">Complaint ID</th><th><font size="" color="black">Username</th>'
 output+='<th><font size="" color="black">Complaint Details</th>'
 output+='<th><font size="" color="black">Complaint Status</th>'
 output+='<th><font size="" color="black">Complaint Date</th>'
 output+='<th><font size="" color="black">Delete Complaint</th></tr>'
 con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root', password = 'root',
database = 'complaint',charset='utf8')
```

```
 with con:
 cur = con.cursor()
 cur.execute("select * FROM user_complaint where username =
'"+uname+"'")
 rows = cur.fetchall()
 for row in rows:
 output+='<td><font size="" color="black">'+str(row[0])+'</td>'
 output+='<td><font size="" color="black">'+row[1]+'</td>'
 output+='<td><font size="" color="black">'+row[2]+'</td>'
 output+='<td><font size="" color="black">'+row[3]+'</td>'
 output+='<td><font size="" color="black">'+row[4]+'</td>'
 output+='<td><a href=\'DeleteComplaint?t1='+str(row[0])+'\'><font
size=3
color=black>Delete</font></a></td></tr>'
 output+= "</table></br></br>"
 context= {'data':output}
 return render(request,
'UserScreen.html', context)
def UploadAudio(request):
 if request.method == 'GET':

return render(request, 'UploadAudio.html', {})
def TextComplaint(request):
 if request.method == 'GET':
```

```
 return render(request, 'TextComplaint.html', {})
def TextComplaintAction(request):
 if request.method == 'POST':
 global uname
 complaint = request.POST.get('t1', False)
 complaint_id = 0
 con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root',
password = 'root',
database = 'complaint',charset='utf8')
 with con:
 cur = con.cursor()
 cur.execute("select max(complaint_id) from user_complaint")
 rows = cur.fetchall()
 for row in rows:
 complaint_id = row[0]
 if complaint_id is not None:
 complaint_id += 1
 else:
 complaint_id = 1
 today = str(date.today())
 db_connection = pymysql.connect(host='127.0.0.1',port = 3306,user
= 'root', password =
'root', database = 'complaint',charset='utf8')
 db_cursor = db_connection.cursor()
```

```
 student_sql_query = "INSERT
user_complaint(complaint_id,username,complaint_details,status,co
mplaint_date)
VALUES('"+str(complaint_id)+"','"+uname+"','"+complaint+"','Pending
','"+today+"')"
```



```
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 print(db_cursor.rowcount, "Record Inserted")
 if db_cursor.rowcount == 1:
 status = "Your Complaint Accepted with Complaint ID =
"+str(complaint_id)+". Our
Admin
will review"
 context= {'data': status}
 return render(request, 'UserScreen.html', context)
def UploadAudioAction(request):
 if request.method == 'POST':
 global uname
 language = request.POST.get('t1', False)
 audio = request.FILES['t2']
 audio_name = request.FILES['t2'].name
```

```
status = "Unable to save your complaint"
fs = FileSystemStorage()
if os.path.exists('ComplaintApp/static/files/'+audio_name):
os.remove('ComplaintApp/static/files/'+audio_name)
filename = fs.save('ComplaintApp/static/files/'+audio_name, audio)
ltype = "en-US"
if language == "Telugu":
ltype = "te-IN"
elif language == "Hindi":
ltype = "hi-IN"
with sr.WavFile('ComplaintApp/static/files/'+audio_name) as source:
audio = recognizer.record(source)
try:
text = recognizer.recognize_google(audio, language=ltype)
except Exception as ex:
text = "unable to recognize"
print(text)
complaint_id = 0
con = pymysql.connect(host='127.0.0.1',port = 3306,user = 'root',
password = 'root',
database = 'complaint',charset='utf8')
with con:
cur = con.cursor()
cur.execute("select max(complaint_id) from user_complaint")
```

```
 rows = cur.fetchall()


for row in rows:
 complaint_id = row[0]
 if complaint_id is not None:
 complaint_id += 1
 else:
 complaint_id = 1
 today = str(date.today())
 db_connection = pymysql.connect(host='127.0.0.1',port = 3306,user
= 'root', password =
'root', database = 'complaint',charset='utf8')
 db_cursor = db_connection.cursor()
 student_sql_query = "INSERT INTO
user_complaint(complaint_id,username,complaint_details,status,co
mplaint_date)
VALUES('"+str(complaint_id)+"','"+uname+"','"+text+"','Pending','"+to
day+"')"
 db_cursor.execute(student_sql_query)
 db_connection.commit()
 print(db_cursor.rowcount, "Record Inserted")
 if db_cursor.rowcount == 1:
 status = "Your Complaint Accepted with Complaint ID =
"+str(complaint_id)+". Our
Admin will review"
```

```
 context= {'data': status}
 return render(request, 'UserScreen.html', context)
os.path.dirname(os.path.dirname(os.path.abspath(__file__))
```