BACKENED CODE:

Python and Django:

from django.shortcuts import render

from django.http import HttpResponse

import smtplib

import speech\_recognition as sr

from email.message import EmailMessage

import pyttsx3

import pymysql

import imaplib

import email

inb1=0

regarr=[]

arr=[]

logn=[]

r = sr.Recognizer()

tts = pyttsx3.Engine()

sender=""

def mic():

with sr.Microphone() as source:

audio\_data = r.record(source,15)

data1 = r.recognize\_google(audio\_data)

return data1.lower()

def long():

with sr.Microphone() as source:

print("Speak now...")

audio\_data = r.record(source,25)

print("Recognizing your text.............")

data1 = r.recognize\_google(audio\_data)

print(data1)

return data1.lower()

def short():

with sr.Microphone() as source:

print("Speak now...")

audio\_data = r.record(source,5)

print("Recognizing your text.............")

data1 = r.recognize\_google(audio\_data)

print(data1)

return data1.lower()

def speak(text):

engine = pyttsx3.init()

voices = engine.getProperty('voices')

engine.setProperty('voice', voices[1].id)

rate = engine.getProperty('rate')

engine.setProperty('rate', rate-20)

engine.say(text)

engine.runAndWait()

def hi1(request):

out={'check':0}

return render(request,'DemoApp/welcome.html',out)

def reg(request):

out={'check':0}

return render(request,'DemoApp/register.html',out)

def log(request):

out={'check':0}

return render(request,'DemoApp/login.html',out)

def enter():

receive = ""

speak("say register to register into this system")

speak("say login to login into this system if you are already a user")

receive = shortInpt()

sum1 = {}

if (receive == "register"):

sum1 = {'count': 2}

speak("You choose to register")

speak("You are now redirected to the registration page")

elif (receive == "login"):

sum1 = {'count': 3}

speak("You choose to login")

speak("You are now redirected to the login page")

else:

speak("You have given incorect voice")

speak("please give correct voice")

sum1=enter()

return(sum1)

def regis(request):

speak("Welcome to voice based email for blind")

speak("In this system input is taken with voice commands")

speak("According to instructions given by the system in the form of voice commands user has to give input with voice commands")

receive=""

sum1=enter()

return render(request, 'DemoApp/welcome.html',sum1)

def openregis(request):

res={'check':0}

return render(request, 'DemoApp/register.html',res)

def emailregis(request):

speak("Enter email id without @gmail.com:")

email=inpt()

z = list(email)

res = ""

for i in z:

if (i != " "):

res = res + i

email=res

email=email+"@gmail.com"

res = res + "@gmail.com"

regarr.append(email)

res={'count':2,'emailp':regarr[0]}

return render(request, 'DemoApp/register.html',res)

def securityregis(request):

speak("Enter security code:")

security=longInpt()

z = list(security)

res = ""

count = 0

for i in z:

if (count == 4):

res = res + " "

count = 0

res = res + i

count = count + 1

security = res

regarr.append(security)

res={'count':3,'emailp': regarr[0],'security1': regarr[1]}

return render(request, 'DemoApp/register.html',res)

def passwdregis(request):

speak("Enter password:")

pas=inpt()

z = list(pas)

res = ""

for i in z:

if (i != " "):

res = res + i

pas=res

regarr.append(pas)

res={'count':4,'emailp': regarr[0],'security1': regarr[1],'passwd1': regarr[2]}

return render(request, 'DemoApp/register.html',res)

def insertdata(request):

d = pymysql.connect(host='localhost', user='root', password='database@0530', database='register')

cur = d.cursor()

n = regarr[0]

s = regarr[1]

p = regarr[2]

cur.execute('insert into form(email,security,password) values(%s,%s,%s)', (n, s, p))

d.commit()

d.close()

speak("Your registration is successfull")

speak("You are now redirected to login page")

res={'check':0}

return render(request, 'DemoApp/login.html',res)

def inpt():

receive = ""

count = 1

while(count):

try:

speak("Speak")

receive = mic()

count = 0

speak("You said")

speak(receive)

except:

speak("I cannot hear your voice speak again")

count=1

speak("Speak yes to confirm or Speak no to say again ")

inp = ""

count = 1

while (count):

try:

speak("Speak")

inp = short()

count = 0

if(inp == 'yes' or inp == 'no' or inp == 'now'):

break

else:

speak("speak correctly")

count=1

except:

speak("i cannot hear your voice speak again")

count=1

if(inp == 'yes'):

return(receive)

elif (inp == 'no' or inp == 'now'):

try:

speak("speak again")

receive=inpt()

except:

pass

return(receive)

def shortInpt():

receive = ""

count = 1

while(count):

try:

speak("Speak")

receive = short()

count = 0

speak("You said")

speak(receive)

except:

speak("I cannot hear your voice speak again")

count=1

speak("Speak yes to confirm or Speak no to say again ")

inp = ""

count = 1

while (count):

try:

speak("Speak")

inp = short()

print("In yes or no:2",inp)

count = 0

if(inp == 'yes' or inp == 'no' or inp == 'now'):

break

else:

speak("speak correctly")

count=1

except:

speak("i cannot hear your voice speak again")

count=1

if(inp == 'yes'):

return(receive)

elif (inp == 'no' or inp == 'now'):

try:

speak("speak again")

receive=inpt()

except:

pass

return(receive)

def longInpt():

receive = ""

count = 1

while(count):

try:

speak("Speak")

receive = long()

count = 0

speak("You said")

speak(receive)

except:

speak("I cannot hear your voice speak again")

count=1

speak("Speak yes to confirm or Speak no to say again ")

inp = ""

count = 1

while (count):

try:

speak("Speak")

inp = short()

count = 0

if(inp == 'yes' or inp == 'no' or inp =='now'):

break

else:

speak("speak correctly")

count=1

except:

speak("i cannot hear your voice speak again")

count=1

if(inp == 'yes'):

return(receive)

elif (inp == 'no' or inp == 'now'):

try:

speak("speak again")

receive=inpt()

except:

pass

return(receive)

def emailvald():

speak("Enter email id without @gmail.com:")

req = inpt()

z = list(req)

res = ""

for i in range(len(z)):

if (z[i] != " "):

res = res + z[i]

req=res

try:

d = pymysql.connect(host='localhost', user='root', password='database@0530', database='register')

cur = d.cursor()

cur.execute('select \* from form where email=%s', (req))

rw=cur.fetchone()

if (rw == None):

speak("The email id given is invalid")

speak("Enter a valid email id")

req = emailvald()

else:

return (req)

except:

pass

return(req)

def pasd():

speak("Enter password:")

req = inpt()

z = list(req)

res = ""

for i in z:

if (i != " "):

res = res + i

req=res

try:

d = pymysql.connect(host='localhost', user='root', password='database@0530', database='register')

cur = d.cursor()

em=logn[0]

cur.execute('select \* from form where email=%s and password=%s', (em,req))

rw=cur.fetchone()

if(rw == None):

speak("The password is invalid")

speak("Enter a correct password")

req=pasd()

else:

return(req)

except:

pass

return(req)

def passwd(request):

print("In password")

pas=pasd()

pas=pas.lower()

logn.append(pas)

res={'count':3,'emailput':logn[0],'passw1':pas}

return render(request, 'DemoApp/login.html',res)

def openvalid(request):

out={'count':4}

speak("Your login details are valid.")

speak("You are redirected to the homepage")

return render(request,'DemoApp/hi.html',out)

def openhome(request):

out={'check':1}

return render(request,'DemoApp/home.html',out)

def opencompose(request):

out={'check':1,'Sendr':logn[0]}

global inb1

inb1=0

return render(request,'DemoApp/compose123.html',out)

def innercompose(request):

speak("Now You can Compose a new mail")

speak("To whom to do you want to send this email")

res1 = inpt()

z = list(res1)

res = ""

for i in z:

if (i != " "):

res = res + i

res=res+"@gmail.com"

arr.append(res)

out = {'count': 1,'sendr':logn[0],'recev':arr[0]}

return render(request, 'DemoApp/compose123.html', out)

def innersub(request):

speak("Enter the subject of this mail")

res=inpt()

arr.append(res)

out = {'count': 2,'sendr':logn[0],'recev':arr[0],'subj':arr[1]}

return render(request, 'DemoApp/compose123.html', out)

def innerbod(request):

speak("Enter the body of this mail")

res=longInpt()

arr.append(res)

send\_mail(logn[0],arr[0],arr[1],arr[2])

out = {'count': 10,'sendr':logn[0],'recev':arr[0],'subj':arr[1],'bdy':arr[2]}

return render(request, 'DemoApp/compose123.html', out)

def last(request):

speak("Your Email is successfully sent")

speak("Now you are redirected to the homepage")

out = {'count': 3,'sendr':logn[0],'recev':arr[0],'subj':arr[1],'bdy':arr[2]}

return render(request, 'DemoApp/compose123.html', out)

def send\_mail(sender,receiver, subject, message):

rw=''

try:

d = pymysql.connect(host='localhost', user='root', password='database@0530', database='register')

cur = d.cursor()

cur.execute('select security from form where email=%s', (sender))

rw=cur.fetchone()

a = list(rw)

rw = a[0]

except:

pass

server = smtplib.SMTP("smtp.gmail.com", 587)

server.starttls()

pas=rw

print(rw)

print(pas)

server.login(sender, pas)

email = EmailMessage()

email["From"] = sender

email["To"] = receiver

email["Subject"] = subject

email.set\_content(message)

server.send\_message(email)

def openinbox(request):

print("In hi1")

out={'count':1}

global inb1

inb1=0

return render(request,'DemoApp/inbox123.html',out)

def openlogout(request):

print("In hi1")

out={'check':1}

return render(request,'DemoApp/logout123.html',out)

def instn():

speak("Say send mail to compose a mail")

speak("say open inbox to go to inbox")

speak("say exit to logout from mail")

speak("say repeat to hear the instructions again")

pas = ""

pas = shortInpt()

if(pas == 'repeat'):

pas=instn()

else:

return(pas)

return(pas)

def checkvalid():

pas = instn()

res = {}

if (pas == 'send mail' or pas == 'sent mail' or pas == 'centmail'):

res = {'count': 1}

speak("You choose to compose a mail")

speak("You are now redirected to compose a new mail")

elif (pas == 'open inbox'):

res = {'count': 2}

speak("You choose to check the inbox")

speak("You are now redirected to check the inbox section")

elif (pas == 'exit'):

res = {'count': 3}

speak("You choose to logout from the mail")

speak("You are now logged out from the mail")

else:

speak("You have given incorect voice")

speak("please give correct voice")

res = checkvalid()

return(res)

def homepg(request):

speak("You are now in the home page")

speak("Here you can compose an email,you can check inbox you can logout from the website ")

res = checkvalid()

return render(request, 'DemoApp/home.html',res)

def get\_email\_body\_time\_wise():

rw = ''

try:

d = pymysql.connect(host='localhost', user='root', password='database@0530', database='register')

cur = d.cursor()

cur.execute('select security from form where email=%s', (logn[0]))

rw = cur.fetchone()

a = list(rw)

rw = a[0]

except:

pass

mailbox = imaplib.IMAP4\_SSL('imap.gmail.com', 993)

mailbox.login(logn[0], rw)

mailbox.select('INBOX')

\_, uids = mailbox.search(None, 'ALL')

emails = []

for uid in uids[0].split():

\_, data = mailbox.fetch(uid, '(RFC822)')

raw\_email = data[0][1]

email\_message = email.message\_from\_bytes(raw\_email)

subject = email\_message['Subject']

sender = email.utils.parseaddr(email\_message['From'])[1]

body = ""

if email\_message.is\_multipart():

for part in email\_message.walk():

if part.get\_content\_type() == 'text/plain':

body = part.get\_payload(decode=True).decode('utf-8')

break

else:

body = email\_message.get\_payload(decode=True).decode('utf-8')

email\_date = email.utils.parsedate\_to\_datetime(email\_message['Date'])

emails.append({

'subject': subject,

'sender': sender,

'body': body,

'date': email\_date

})

emails.sort(key=lambda x: x['date'], reverse=True)

mailbox.close()

mailbox.logout()

return emails

emails=[]

def mailData():

global inb1

global emails

j = 0

for email in emails:

if (j == inb1):

speak("date is")

dt = email['date']

dt = str(dt)

dt = list(dt)

n = len(dt)

ar = dt[0:(n - 6)]

s = ''

for i in ar:

s = s + i

speak(s)

print("Sender:", email['sender'])

speak("Subject of the mail:")

speak(email['subject'])

print("Subject:", email['subject'])

print("Date:", email['date'])

speak("Body of the mail is ")

speak(email['body'])

print("Body:", email['body'])

print("==========================")

speak("Now we taking you to read next emails")

inb1=inb1+1

return(1)

j=j+1

def invit():

global emails

emails=get\_email\_body\_time\_wise()

speak("You are now in the inbox page")

speak("You can now read emails ")

return(1)

def speakInbox(request):

op=mailData()

out={'count': 1}

return render(request, 'DemoApp/inbox123.html', out)

def readInbox(request):

global inb1

global emails

if(inb1 == 0):

ret=invit()

i=0

for email in emails:

if(i == inb1):

speak("The sender of this email is")

speak(email['sender'])

speak("speak go if you want read this email")

speak("speak next to read next email")

speak("speak back to go back to homepage")

ele = shortInpt()

if (ele == "next"):

inb1=inb1+1

i=i+1

continue

if(ele == 'back'):

out={'count':2}

return render(request, 'DemoApp/inbox123.html', out)

if(ele == 'go'):

dt = email['date']

dt = str(dt)

dt = list(dt)

n = len(dt)

ar = dt[0:(n - 6)]

s = ''

for i in ar:

s = s + i

print("Sender:", email['sender'])

print("Subject:", email['subject'])

print("Date:", email['date'])

print("Body:", email['body'])

print("==========================")

out={'count':10,'Sender':email['sender'],'Date':s,'Subject':email['subject'],'Body':email['body']}

return render(request, 'DemoApp/inbox123.html', out)

i=i+1

speak("You have read all emails.")

speak("Thank you")

out={'check':2}

return render(request,'DemoApp/inbox123.html',out)