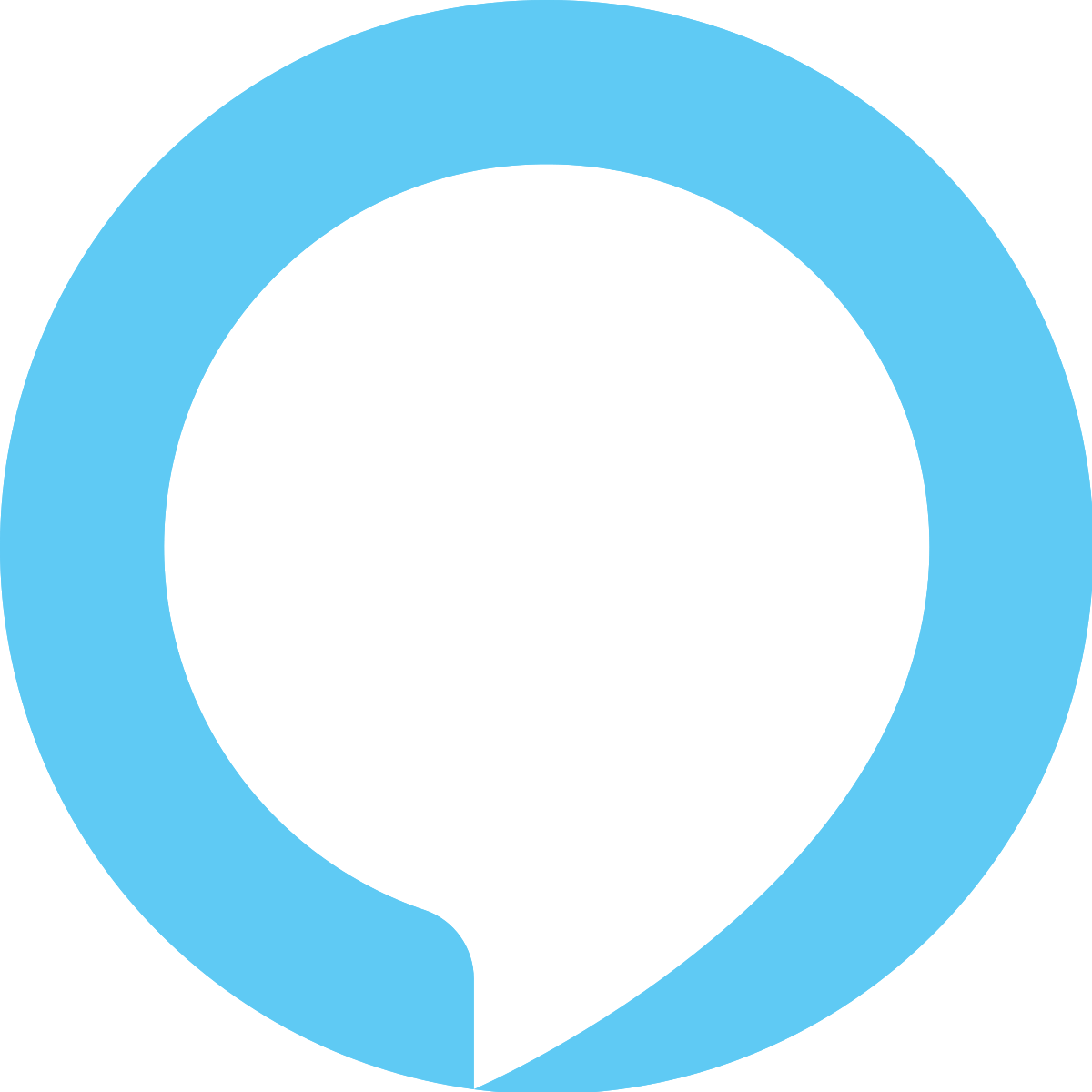
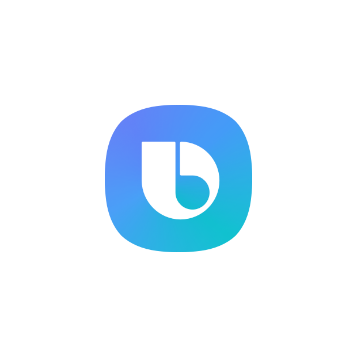
 

FRIDAY – THE VIRTUAL ASSISTANT



CONTENTS

* ACKNOWLEDGEMENT
* ABSTRACT

* REQUIREMENT ANALYSIS:
* HARDWARE REQUIREMENTS
* SOFTWARE REQUIREMENTS
* APPLICATIONS
* DESIGN:
* BLOCK DIAGRAM
* EXPLANATION
* FLOW CHART:
* PROJECT FLOW
* ALGORITHEM
* SOURCE CODE
* SAMPLE OUTPUT
* FUTURE ENHANCEMENT
* BIBLIOGRAPHY

**ACKNOWLEDGEMENT:**

In an order to complete this project, I had to take the help and guideline of a few respected people.

Apart from the efforts of me and my team, the success of any project depends largely on the encouragement and guidelines of many others. I take this opportunity to express my gratitude to the people who have been instrumental in the successful completion of this project.

I express deep sense of gratitude to almighty God for giving me strength for the successful completion of the project.

I express my heartfelt gratitude to my parents for constant encouragement while carrying out this project.

I gratefully acknowledge the contribution of the individuals who contributed in bringing this project up to this level, who continues to look after me despite my flaws.

I sincerely thank my computer science teacher ‘Ms. Ananthy’ who encouraged me to do this project.

**INTRODUCTION:**

In today’s era almost all tasks are digitalized. We have Smartphone in hands and it is nothing less than having world at your finger tips. These days we aren’t even using fingers. We just speak of the task and it is done. There exist systems where we can say Text Dad, “I’ll be late today.” And the text is sent. That is the task of a Virtual Assistant. It also supports specialized task such as booking a flight, or finding cheapest book online from various ecommerce sites and then providing an interface to book an order are helping automate search, discovery and online order operations.

We present you a virtual assistant who can perform these tasks very efficiently and elegantly. ’Friday’ a virtual assistant we have designed and work on for quite sometime now it can perform any task given to it that is the things we programmed.

We have created ‘Friday’ as user friendly assistant and as a powerful virtual assistant. First and foremost, basically you would need a FRIDAY account to access the virtual assistant.

We have done ‘Friday’ as a team project.

**ABSTRACT:**

The aim of our project ‘Friday’ is to make a better virtual assistant for all the users who have a Friday account.

Till now we have created ‘FRIDAY’ for these goals.

They are:

* TO CREATE A FRIDAY ACCOUNT.
* TO SCREEN MIRROR USER’S PHONE USING USB CABLE.
* TO FIND USER’S PHONE USING THERE GOOGLE ACCOUNT.
* TO OPEN OTHER APPLICATION.
* TO CALCULATE MATHAMATICAL OPERATION.
* TO COUNT NUMBERS.
* TO ACT AS A CHATTER BOX.
* TO MAKE A NOTE FOR THE USER.
* TO GENERATE RANDOM NUMBER.
* TO DISPLAY TODAY’S DATE, DAY, TIME, MONTH, YEAR.
* TO SAY TODAY’S WEATHER.
* TO OPEN WEBSITES.
* TO TURN OFF THE USER’S COMPUTER.
* TO MAKE A NOTE
* TO SEE ALL THE PREVIOUS NOTES

That’s all we were able to do during this 7 months. We will try our best to add other features in future.

**REQUIREMENT ANALYSIS:**

1. **HARDWARE REQUIREMENTS:**

* BASIC LEVEL CENTRAL PROCESSING UNIT [CPU].
* 4 GB RAM
* KEYBOARD

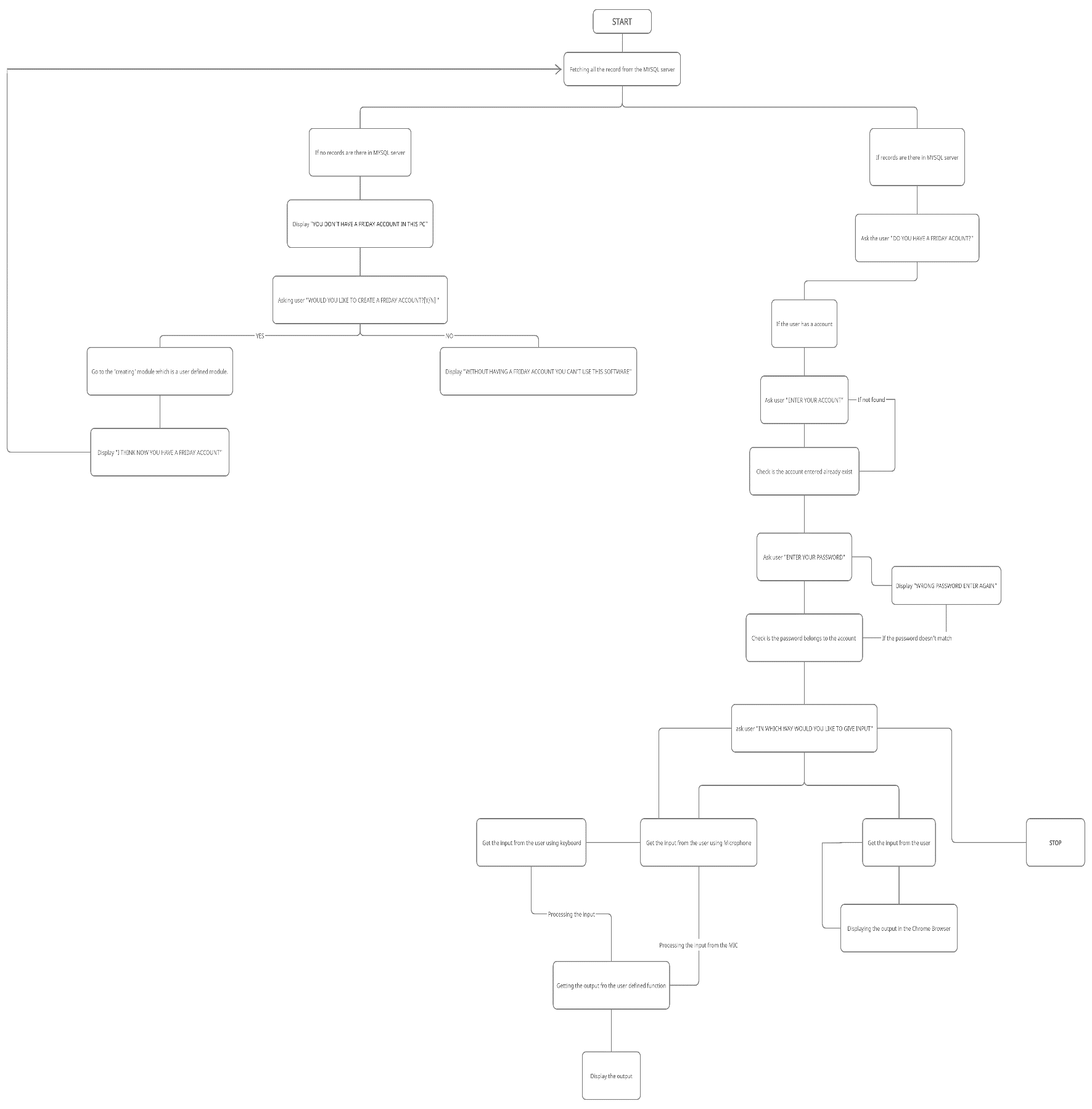
1. **SOFTWARE REQUIREMENTS:**

* WINDOWS 7 AND ABOVE OPERATING SYSTEM.
* PYTHON 3.7 AND LATER.
* MYSQL
* CHROME BROWSER.
* SCRCPY

**APPLICATIONS:**

* This software is made with the aim to serve people by being their virtual assistance.
* This software can be use by physically challenged people.
* This software can be used as a home assistant to open an application in the PC, to screen mirror user’s smartphone, to make a reminder, to play YouTube video and etc.
* This software can be used my people who all are tried of give inputs using your keyboard.
* This software can be used to learn new innovative things

**DESIGN**:

1. **BLOCK DIAGRAM:** 
2. **EXPLINATION:**

* On running this project first, the program will fetch all records from the My SQL local server.
* If the records fetched are zero then the program will print “YOU DON’T HAVE A FRIDAY ACCOUNT IN THIS PC” and then the program will ask the user “WOULD YOU LIKE TO CREATE A FRIDAY ACCOUNT?”.
* If user type’s “YES” then the program will run “creating” which is a user defined module, which will help in creating a “FRIDAY ACCOUNT”.
* If user type’s “NO” when the program will stop and it will print

“WITHOUT HAVING A FRIDAY ACCOUNT YOU CAN’T USE THIS PROGRAM” and the program will terminate

* If there are records already exist then the program will ask the user “DO YOU HAVE A FRIDAY ACCOUNT? [Y/N]:”
* If the answer is “YES”:

1. the program will ask “ENTER YOUR ACCOUNT: “and the program will check there is a Friday account already exist.
2. If the Friday account doesn’t exist then the program will print “NO ACCCOUNT FOUND ENTER YOUR ACCOUNT AGAIN”.
3. If account exist the then program will ask the user “ENTER YOUR PASSWORD: “.
4. If the password belongs to the account holder, then the main program will run.
5. After the sign up and log in work the program will ask “IN WHICH WAY WOULD YOU LIKE TO GIVE INPUT:
6. TEXT, VOICE, HYPERSEARCH, OR DO YOU WANT TO GIVE EXIT THE PROGRAM.
7. After getting the input method the program get operation which are given by the user.
8. The program will not stop unless the user wants to stop.

* If the answer is “NO”:

1. “The program will ask would you like to create a Friday account? [y/n]:”

* If the answer is “YES”:

1. The program will run ‘creating’ which is a user defined module which will help user to create a Friday account and after that the program will run again from first.

* If the answer is “NO”:

1. Then the program will print “WITHOUT HAVING A FRIDAY ACCOUNT YOU CAN’T USE THIS PROGRAM” and the program will terminate.

**ALGORITHM:**

Step 1: Start

Step 3: read value record

Step 4: if record == []:

Step 5: display “YOU DON’T HAVE A FRIDAY ACCOUNT IN THIS PC”

Step 6: declare a variable cr

Step 7: display “WOULD YOU LIKE TO CREATE A FRIDAY ACCOUNT[Y/N]:” and get input from the user using cr

Step 8: If cr == ‘YES’:

Step 9: declare a variable account

Step 10: display “ENTER YOUR ACCOUNT:” and get input using account

Step 11: Check is there any account already exist if not then ask user to enter the account again

Step 12: declare a variable password

Step 13: display “ENTER YOUR PASSWORD:” and get input using password

Step 14: check does the password belongs to account holder, if doesn’t ask the password again.

Step 15: declare a variable inputm

Step 16: display “IN WHICH WAY WOULD YOU LIKE TO GIVE INPUT:” and get the input using the variable inputm

Step 17: if inputm == “EXIT”:

Step 18: stop

Else:

Step 18:continue with the operation which user has given

Else:

Step 9: display “WITHOUT HAVING A FRIDAY ACCOUNT YOU CAN’T USE THIS PROGRAM”

Step 10: stop

Else:

Step 5: declare variable conf

Step 6: display “DO YOU HAVE A FRIDAY ACCOUNT?” and get input from the user using conf

Step6: if conf == ‘YES’:

Step 7: go to the ‘creating’ module.

Else:

Step 7: declare a variable conf2

Step 8: display “WOULD YOU LIKE TO CREATE A FRIDAY ACCOUNT?” and get input from the user using conf2

Step 9: if conf2 == ‘YES’:

Step 10: go to the ‘creating’ module.

Else:

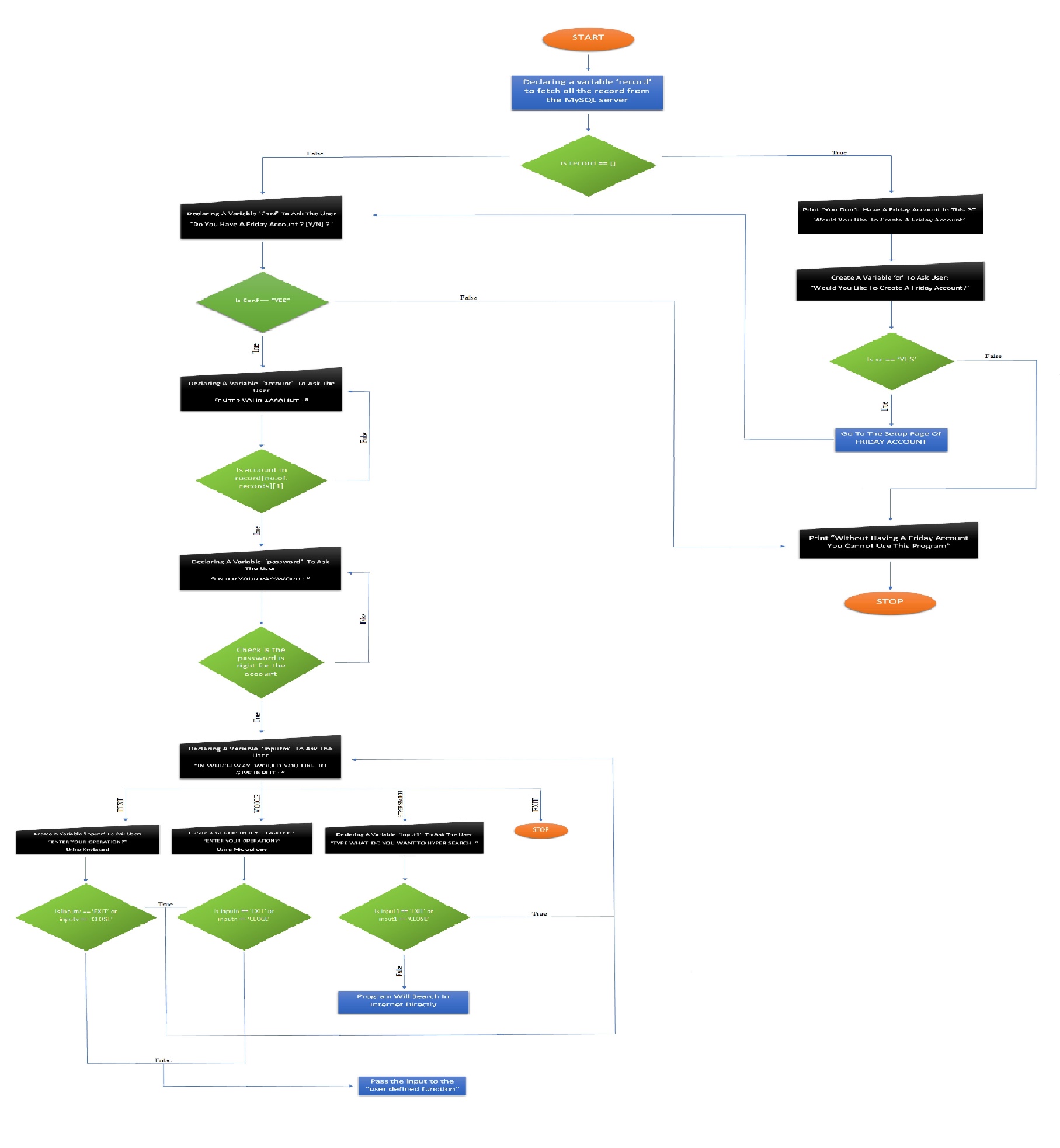
Step 10: display “WITHOUT HAVING A FRIDAY ACCOUNT YOU CAN’T USE THIS PROGRAM”

Step 11: stop

Step 5: display “ERROR”

Step 6: stop

**FLOW CHART:**



**SOURCE CODE:**

MAIN\_FRIDAY.py

try:

print()

print("===============","FRIDAY","===============")

import pyttsx3

import pyaudio

import pyjokes

import pywhatkit

import speech\_recognition as sr

import enchant as search

import mysql.connector as ms

import wikipedia

from math import \*

listener = sr.Recognizer()

engine = pyttsx3.init()

voices = engine.getProperty('voices')

engine.setProperty('voices', voices[0].id)

engine.runAndWait()

def speak(process):

engine.say(process)

engine.runAndWait()

def action():

try:

with sr.Microphone() as voice\_input:

print('I AM LISTENING......', end=' = ')

temp\_input = listener.listen(voice\_input)

process = listener.recognize\_google(temp\_input)

print(process)

return process

except:

print()

print('ERROR M1: THERE IS A PROBLEM IN YOUR MICROPHONE')

def check(inputa):

dictionary = search.Dict('en\_us')

input2 = inputa.lower().split()

t = 0

if inputa == 'NONE':

print("SORRY CAN YOU SPEAK AGAIN")

print()

else:

for i in input2:

if dictionary.check(i):

t += 1

else:

pass

if len(input2) == t:

pywhatkit.search(inputa)

else:

print('SORRY BUT I AM UNABLE TO UNDERSTAND.')

def main\_action(inputa, v, name, age, phno, dob):

if 'FUN FACT' in inputa or 'INTERESTING FACTS' in inputa or 'RANDOM FACT' in inputa or 'AMAZING FACT' in inputa:

print()

import randfacts

temp = randfacts.get\_fact()

print(temp)

if v == 1:

speak(temp)

elif 'RANDOM NUMBER' in inputa or 'PICK A RANDOM NUMBER' in inputa:

import MODULE.random1

temp = MODULE.random1.random1()

if v == 1:

speak(temp)

elif '/' in inputa or '\*' in inputa or '-' in inputa or '+' in inputa or 'sin' in inputa or 'cos' in inputa or 'tan' in inputa:

try:

inputa = inputa.replace('YOU', '')

inputa = inputa.replace('CAN', '')

inputa = inputa.replace('HELLO', '')

inputa = inputa.replace('FRIDAY', '')

inputa = inputa.replace('HEY', '')

inputa = inputa.replace('HI', '')

inputa = inputa.replace('CALCULATE', '')

inputa = inputa.replace('CALCULATOR', '')

cal1 = str(inputa).lower()

cal1 = eval(cal1)

print(cal1)

if v == 1:

speak(cal1)

except:

check(inputa)

elif 'PRESENT DAY' in inputa or 'TODAY DAY' in inputa or "PRESENTS DAY" in inputa or "TODAYS DAY" in inputa or "PRESENTS DAYS" in inputa or "TODAYS DAYS" in inputa or "PRESENT'S DAYS" in inputa or "TODAY'S DAYS" in inputa or "PRESENT DAYS" in inputa or "TODAY DAYS" in inputa or "PRESENT'S DAY" in inputa or "TODAY'S DAY" in inputa:

import MODULE.tdmy\_edit

temp = MODULE.tdmy\_edit.day()

if v == 1:

speak(temp)

elif 'PRESENT DATE' in inputa or 'PRESENT MONTH' in inputa or 'PRESENT YEAR' in inputa or 'TODAY DATE' in inputa or 'TODAY MONTH' in inputa or 'TODAY YEAR' in inputa:

import MODULE.tdmy\_edit

temp = MODULE.tdmy\_edit.today()

if v == 1:

speak(temp)

elif 'PRESENT WEEK' in inputa or 'TODAY WEEK' in inputa or 'PRESENT WEEKS' in inputa or 'TODAY WEEKS' in inputa or "PRESENT'S WEEK" in inputa or "TODAY'S WEEK" in inputa:

import MODULE.tdmy\_edit

temp = MODULE.tdmy\_edit.day()

if v == 1:

speak(temp)

elif 'TIME' in inputa or 'HOUR' in inputa or 'MINUTE' in inputa or 'PRESENT TIME' in inputa or 'PRESENT HOUR' in inputa or 'PRESENT MINUTE' in inputa or 'TODAY TIME' in inputa or 'TODAY HOUR' in inputa or 'TODAY MINUTE' in inputa:

import MODULE.tdmy\_edit

temp = MODULE.tdmy\_edit.time()

if v == 1:

speak(temp)

elif 'COUNT' in inputa:

import MODULE.count

temp = MODULE.count.count()

if v == 1:

speak(temp)

elif 'WWW' in inputa or 'COM' in inputa or 'ORG' in inputa or 'EDU' in inputa or 'GOV' in inputa:

inputa = inputa.replace('HI', '')

inputa = inputa.replace('HEY', '')

inputa = inputa.replace('HELLO', '')

inputa = inputa.replace('FRIDAY', '')

inputa = inputa.replace('CAN', '')

inputa = inputa.replace('YOU', '')

inputa = inputa.replace('OPEN', '')

inputa = inputa.strip()

import MODULE.website1

MODULE.website1.website2(str(inputa))

elif 'LOVE YOU' in inputa:

import MODULE.hlmv

temp = MODULE.hlmv.love()

if v == 1:

speak(temp)

elif 'MARRY YOU' in inputa:

import MODULE.hlmv

temp = MODULE.hlmv.marry()

if v == 1:

speak(temp)

elif 'OPEN' in inputa:

inputa = inputa.replace('HI', '')

inputa = inputa.replace('HEY', '')

inputa = inputa.replace('HELLO', '')

inputa = inputa.replace('FRIDAY', '')

inputa = inputa.replace('CAN', '')

inputa = inputa.replace('YOU', '')

input2 = inputa.replace('OPEN', '')

input2 = input2.strip()

input2 = input2.lower()

import MODULE.app

MODULE.app.open1(input2)

elif "MY NAME" in inputa:

print("YOUR NAME IS", name)

if v == 1:

speak(name)

elif 'SCREEN MIRROR' in inputa:

import MODULE.app

MODULE.app.smapp()

elif 'MAKE A NOTE' in inputa or 'MAKE NOTE' in inputa or 'MAKE A REMINDER' in inputa or 'MAKE REMINDER' in inputa:

import MODULE.note

MODULE.note.write\_note(account)

elif 'MY NOTE' in inputa or 'SHOW REMINDER' in inputa or 'SHOW MY REMINDER' in inputa or 'REMINDER' in inputa or 'DISPLAY NOTE' in inputa or 'DISPLAY MY NOTE' in inputa:

import MODULE.note

MODULE.note.read\_note(account)

elif "MY AGE" in inputa:

print("YOUR AGE IS", age)

if v == 1:

speak(age)

elif "MY BIRTHDAY" in inputa or "MY DATE OF BIRTH" in inputa or "MY DOB" in inputa:

print('YOUR DATE OF BIRTH IS:', dob)

elif "MY PHONE NUMBER" in inputa or "MY PH NO" in inputa or "MY PH" in inputa or "MY PHNO" in inputa:

print("YOUR PHONE NUMBER IS", phno)

if v == 1:

speak(phno)

elif "WHO" in inputa:

inputa = inputa.replace('HEY', '')

inputa = inputa.replace('HI', '')

inputa = inputa.replace('HELLO', '')

inputa = inputa.replace('FRIDAY', '')

inputa = inputa.replace('CAN', '')

inputa = inputa.replace('YOU', '')

temp = inputa.replace("WHO IS THE", "")

temp = inputa.replace("WHO IS", "")

temp = str(wikipedia.summary(temp, 10))

temp = temp.replace(".", ".\n")

print(temp)

if v == 1:

speak(temp)

elif "WHAT" in inputa:

pywhatkit.search(inputa)

print("YOUR ARE SEE THE RESULT IN A NEW WINDOW")

elif 'FIND MY PHONE' in inputa or 'SEARCH MY PHONE' in inputa or 'FIND MY DEVICE' in inputa or 'SEARCH MY DEVICE' in inputa:

import MODULE.app

MODULE.app.find1()

elif 'JOKE' in inputa:

print()

temp = str(pyjokes.get\_joke())

print(temp)

if v == 1:

speak(temp)

elif 'PLAY' in inputa:

inputa = inputa.replace('HEY', '')

inputa = inputa.replace('HI', '')

inputa = inputa.replace('HELLO', '')

inputa = inputa.replace('FRIDAY', '')

inputa = inputa.replace('CAN', '')

inputa = inputa.replace('YOU', '')

input2 = inputa.replace('PLAY', '')

pywhatkit.playonyt(input2)

elif 'WEATHER' in inputa:

import MODULE.weather

we = MODULE.weather

elif 'SIRI' in inputa or 'ALEXA' in inputa or 'GOOGLE' in inputa or 'BIXBY' in inputa:

import MODULE.hlmv

temp = MODULE.hlmv.voice1()

if v == 1:

speak(temp)

elif inputa == 'HI' or inputa == 'HELP' or inputa == 'HELLO' or inputa == 'HI FRIDAY' or inputa == 'HELP FRIDAY' or inputa == 'HELLO FRIDAY':

import MODULE.hlmv

temp = MODULE.hlmv.hi()

if v == 1:

speak(temp)

elif 'SHUTDOWN MY PC' in inputa or 'SHUTDOWN THE PC' in inputa or 'SHUTDOWN PC' in inputa or 'SHUT PC' in inputa or 'SHUT MY PC' in inputa or 'SHUT THE PC' in inputa:

import os

os.system('shutdown /s')

elif 'RESTART MY PC' in inputa or 'RESTART THE PC' in inputa or 'RESTART PC' in inputa:

import os

os.system('shutdown /r')

elif 'SHUTDOWN' in inputa:

import os

os.system('shutdown /s')

else:

check(inputa)

while True:

connector = ms.connect(host='localhost', user='root', passwd='rizwaan')

cursor = connector.cursor()

cursor.execute('create database if not exists friday')

cursor.execute('use friday')

cursor.execute(

"create table if not exists friday(userid int primary key,account varchar(100) unique,name varchar(100) not null,date\_of\_birth date not null,age int not null,phone\_number bigint not null,password varchar(100) not null)")

cursor.execute('select \* from friday')

record = cursor.fetchall()

l = cursor.rowcount

print()

if record == []:

idk1 = 0

print("""YOU DO NOT HAVE A FRIDAY ACCOUNT IN THIS PC.

TO USE THIS APPLICATION YOU NEED TO HAVE A FRIDAY ACCOUNT.""")

while True:

print()

cr = input("""WOULD YOU LIKE TO CREATE A FRIDAY ACCOUNT:

1.YES

2.NO

ANSWER: """).strip().upper()

if 'YES' in cr or cr == 'S' or cr == '' or cr == '1':

import MODULE.creating

MODULE.creating.account()

print()

print("I THINK NOW YOU ARE GOOD TO GO WITH YOUR 'FRIDAY ACCOUNT'")

print()

break

elif 'NO' in cr or cr == 'N' or 'NOPE' in cr or cr == '0' or cr == '2':

print()

print("WITHOUT HAVING A FRIDAY ACCOUNT YOU CAN'T USE THIS PROGRAM")

idk1 = 1

break

else:

print("KINDLY GIVE PROPER INPUT")

continue

if idk1 == 1:

break

elif idk1 == 0:

continue

else:

idk2 = 0

while True:

if idk2 == 0:

conf = input("""DO YOU HAVE A 'FRIDAY ACCOUNT'?

1. YES

2. NO

ANSWER: """).strip().upper()

if 'YES' in conf or conf == 'S' or conf == '' or conf == '1':

break

elif 'NO' in conf or conf == 'N' or 'NOPE' in conf or conf == '0' or conf == '2':

while True:

print()

conf2 = input("""WOULD YOU LIKE TO HAVE A 'FRIDAY ACCOUNT'?

1. YES

2. NO

ANSWER: """).strip().upper()

if 'YES' in conf2 or conf2 == 'S' or conf2 == '' or conf2 == '1':

print()

print("YOUR ARE IN THE SETUP PAGE OF 'FRIDAY ACCOUNT':")

import MODULE.creating

MODULE.creating.account()

print()

print("I THINK NOW YOU ARE GOOD TO GO WITH YOUR 'FRIDAY ACCOUNT'")

break

elif 'NO' in conf2 or conf2 == 'N' or conf2 == 'NOPE' or conf2 == '0' or conf2 == '2':

idk2 = 1

print()

print("SORRY WITHOUT HAVING A 'FRIDAY ACCOUNT' YOU CAN'T USE THIS PROGRAM")

break

else:

print("KINDLY GIVE PROPER INPUT")

continue

else:

print("KINDLY GIVE PROPER INPUT")

continue

else:

break

if idk2 == 1:

break

while True:

print()

account = input('ENTER YOUR ACCOUNT:').lower()

account = account.replace('@', '')

account = account.replace('@friday.com', '')

account = account.replace('@friday', '')

account = account.replace('.com', '')

account = account + '@friday.com'

t = 0

for tp in range(0, l):

if record[tp][1] == account:

t = 1

break

else:

continue

if t == 1:

print("OK THEN")

break

else:

print("NO ACCOUNT FOUND.")

print('TRY ENTERING YOUR ACCOUNT NAME AGAIN:')

continue

while True:

print()

password = input("ENTER YOU ACCOUNT's PASSWORD:")

cursor.execute("select \* from friday where account = '%s'" % account)

f = cursor.fetchone()

if f[6] == password:

print('OK TO GO WITH YOUR PASSWORD')

break

else:

print('SORRY YOUR PASSWORD IS WRONG')

continue

print()

print("IF DO YOU WANT TO SEARCH IN GOOGLE DIRECTLY THE TYPE 'HYPER SEARCH'")

while True:

t = 0

name = record[tp][2]

age = record[tp][4]

phno = record[tp][5]

dob = str(record[tp][3])

inputm = input("""IN WHICH METHOD WOULD YOU LIKE TO GIVE:

1) TEXT

2) VOICE

3) HYPER SEARCH

4) OR DO YOU WANT TO EXIT THE PROGRAM

ANSWER: """).upper().strip()

while True:

if "TEXT" in inputm or inputm == '1':

v = 0

print()

inputn = input('ENTER YOUR OPERATION: ').upper().strip()

try:

if inputn == 'EXIT' or inputn == 'CLOSE':

break

else:

main\_action(inputn, v, name, age, phno, dob)

except:

check(inputn)

elif "VOICE" in inputm or inputm == '2':

v = 1

print()

inputv = str(action()).upper()

try:

if inputv == 'EXIT' or inputv == 'CLOSE':

break

else:

main\_action(inputv, v, name, age, phno, dob)

enter\_pass = input("PRESS 'ENTER KEY' TO SPEAK AGAIN:").replace(enter\_pass, '')

if enter\_pass == '':

continue

except:

pass

elif 'HYPER SEARCH' in inputm or inputm == '3':

print()

input1 = input('WHAT DO YOU WANT TO SEARCH: ').lower()

if input1 == 'exit' or input1 == 'close':

break

else:

pywhatkit.search(input1)

elif "EXIT" in inputm or inputm == '4' or "CLOSE" in inputm:

t = 1

print()

print("SEE YOU NEXT TIME " + "'" + str(name).upper() + "'")

print("BYE BYE.")

break

else:

print()

print('''WRONG INPUT METHOD.

PLEASE ENTER AGAIN.''')

break

if t == 1:

break

if t == 1:

break

except ModuleNotFoundError:

print()

x = open("G:\\My Drive\\FRIDAY\\SETUP\\SETUP.txt",'r')

print(x.read())

CREATING A FRIDAY ACCOUNT

import random

from datetime import datetime

import mysql.connector as ms

def account():

try:

connector = ms.connect(host='localhost', user='root', passwd='rizwaan')

cursor = connector.cursor()

cursor.execute('create database if not exists friday')

connector.commit()

connector.close()

connector = ms.connect(host='localhost', user='root', passwd='rizwaan', database='friday')

cursor = connector.cursor()

cursor.execute(

'create table if not exists friday(userid int primary key,account varchar(100) unique,name varchar(100) not null,date\_of\_birth date not null,age int not null,phone\_number bigint not null,password varchar(100) not null)')

connector.commit()

cursor.execute('select \* from friday')

x = cursor.fetchall()

l = cursor.rowcount

while True:

while True:

print()

name = input('ENTER YOUR NAME:').strip().lower()

if '!' in name or '@' in name or '#' in name or '$' in name or '%' in name or '^' in name or '&' in name or '\*' in name or '(' in name or ')' in name or '\_' in name or '=' in name or '/' in name or '-' in name or '+' in name or '<' in name or '>' in name or '?' in name or '\\' in name or ',' in name or "'" in name or '"' in name or ':' in name or ';' in name or '{' in name or '}' in name or '[' in name or ']' in name or '|' in name or '~' in name or '`' in name:

print("PLEASE DON'T ENTER ANY SPECIAL CHARACTER IN YOUR NAME.")

continue

else:

break

while True:

print()

account = input('CREATE YOUR ACCOUNT:').strip()

account = account.replace('@', '')

account = account.replace('@friday.com', '')

account = account.replace('@friday', '')

account = account.replace('.com', '')

if '(' in account or ')' in account or '=' in account or '<' in account or '>' in account or '?' in account or '\\' in account or ',' in account or "'" in account or '"' in account or ':' in account or ';' in account or '{' in account or '}' in account or '[' in account or ']' in account or '|' in account or '~' in account or '`' in account:

print("SORRY YOU CAN'T USE FEW SPECIFIC CHARACTERS.")

continue

elif account == '':

print("KINDLY FILL YOUR DETAILS PROPERLY")

else:

if l == 0:

account = account + '@friday.com'

break

else:

account = account + '@friday.com'

t = 0

for tp in range(0, l):

if x[tp][1] == account:

print('''ALREADY THERE IS A ACCOUNT ALREADY EXIST

KINDLY HAVE SOME OTHER ACCOUNT NAME''')

t = 0

break

else:

t = 1

continue

if t == 0:

continue

else:

break

while True:

print()

iy = int(input("ENTER YOUR 'YEAR OF BIRTH': "))

year1 = datetime.now()

year1 = str(year1).split()

year1 = year1[0].replace('-', ' ').split()

year1 = int(year1[0])

if len(str(iy)) < 4 or len(str(iy)) > 4:

print('PLEASE ENTER YOUR AGE IN 4 DIGIT FORMAT (yyyy).')

continue

else:

if iy > year1:

print('SORRY YOUR ARE NOT FROM', iy, '\n'

'I AM SURE.')

continue

elif iy < year1:

age = year1 - iy

if age >= 10:

print('OK TO GO WITH YOUR AGE.')

break

else:

print('''YOUR AGE IS LESS THEN 10.

YOU CAN'T OPEN YOU FRIDAY ACCOUNT WITH THIS AGE.''')

continue

else:

print('SORRY BUT HOW CAN BOTH THE YEARS BE SAME.')

continue

while True:

print()

month1 = int(input("ENTER YOUR 'MONTH OF BIRTH': "))

if month1 <= 12:

print('OK TO GO WITH YOU MONTH NUMBER.')

break

else:

print('YOUR MONTH OF BIRTH IS NOT PROPER.')

continue

while True:

print()

date1 = int(input("ENTER YOUR 'DATE OF BIRTH': "))

if date1 <= 31:

print("OK TO GO WITH YOUR 'DATE OF BIRTH'.")

break

else:

print("YOUR 'DATE OF BIRTH' IS NOT PROPER")

continue

while True:

print()

phno = int(input('ENTER YOUR THE 10 DIGIT PHONE NUMBER: '))

if len(str(phno)) == 10:

print('OK TO GO WITH THE ', "'", phno, "'", ' AS YOUR PHONE NUMBER.')

break

else:

print('''JUST ENTER 10 DIGIT PHONE NUMBER.

INSTANT YOU HAVE ENTERED:''', phno)

continue

print()

print("THE PASSWORD YOU ENTER IS CASE SENSITIVE")

while True:

print()

password1 = input('ENTER YOUR DESIRED PASSWORD: ')

password2 = input('ENTER YOUR DESIRED PASSWORD TO CONFIRMED:')

if password1 == password2:

password = password1

print('OK TO GO WITH THE PASSWORD.')

break

else:

print('SORRY CAN YOU RE-ENTER YOUR PASSWORD BECAUSE BOTH THE PASSWORD ARE NOT SAME.')

continue

date\_of\_birth = str(iy) + '-' + str(month1) + '-' + str(date1)

while True:

print()

check = """YOUR NAME IS {name}

YOUR ACCOUNT IS {account}

YOUR DATE OF BIRTH IS {date\_of\_birth}

YOUR PHONE NUMBER IS {ph\_no}

YOUR PASSWORD IS {password}

""".format(name=name.upper(), account=account, date\_of\_birth=date\_of\_birth, ph\_no=phno, password=password)

print(check)

w = 0

s = input("""IS THE DETAILS ARE CORRECT [Y/N]: """).upper().strip()

if 'YES' in s or s == 'OK' or s == 'S' or s == '1' or s == 'Y' or s == '':

print('OK THEN')

break

elif 'NO' in s or 'NOPE' in s or s == 'N' or s == '0':

re = input('DO YOU WANT TO ENTER YOUR DETAILS AGAIN: ').upper().strip()

if 'YES' in re or 'OK' in re or re == 'S' or re == '1' or re == 'Y':

w = 1

break

elif 'NO' in re or 'NOPE' in re or re == "N" or re == '0':

print("THEN I WILL PRECEDE WITH YOR DETAILS")

break

else:

print('JUST TYPE [Y/N]')

continue

if x == [] and w == 0:

userid = random.randint(1000, 9999)

insert = "insert into friday values({userid},'{account}','{name}','{date\_of\_birth}',{age},{phone\_number},'{password}')".format(

userid=userid, account=account, name=name, date\_of\_birth=date\_of\_birth, age=age, phone\_number=phno,

password=password)

cursor.execute(insert)

connector.commit()

connector.close()

elif x != [] and w == 0:

while True:

userid = random.randint(1000, 9999)

t = 0

for i in range(0, l):

if x[i][0] == userid:

break

else:

t += 1

continue

if t == 0:

continue

else:

break

insert = "insert into friday values({userid},'{account}','{name}','{date\_of\_birth}',{age},{phone\_number},'{password}')".format(

userid=userid, account=account, name=name, date\_of\_birth=date\_of\_birth, age=age, phone\_number=phno,

password=password)

cursor.execute(insert)

connector.commit()

connector.close()

else:

print()

print("ENTER YOUR DETAIL'S AGAIN BUT PROPERLY")

continue

break

except:

print("""THERE IS A PROBLEM IN THE CREATING YOUR ACCOUNT.

KINDLY TRY LATER""")

OTHER MODULES IN PACKAGE

1. app.py:

def smapp():

print()

input1 = input(' HAVE YOU CONNECTED YOUR PHONE USING AN USB CABLE [Y/N] :').upper()

if input1 == 'Y' or input1 == 'YES':

import os

os.startfile('D:\\RIZWAAN\\True\_Friday\\APPLICATION\\scrcpy\\scrcpy.exe')

print()

print('NOW YOU ARE SCREEN MIRRORING YOUR PHONE.')

elif input1 == 'N' or input1 == 'NO':

print()

print('OK THEN TRY NEXT TIME')

else:

print()

print('''WRONG INPUT

TRY AGAIN LATER''')

def find1():

print()

print('NOTE: MAKE SURE YOU HAVE SIGNED IN THE BROWSER WITH THE SAME GOOGLE ACCOUNT WHICH YOU HAVE SIGNED IN YOU PHONE.')

print()

temp = input('DO YOU HAVE A GOOGLE ACCOUNT SIGNED IN YOUR PHONE: ').upper()

if temp == 'YES' or temp == 'Y' or temp == '1':

import webbrowser

webbrowser.register('chrome',

None,

webbrowser.BackgroundBrowser('C:\\Program Files\\Google\\Chrome\\Application\\chrome.exe'))

webbrowser.get('chrome').open('https://www.google.com/android/find?u=0')

elif temp == 'NO' or temp == 'N' or temp == '0':

print()

print("SORRY BUT WE CAN'T HELP YOU WITHOUT HAVING A GOOGLE ACCOUNT SIGNED IN YOUR PHONE")

else:

print()

print('WRONG INPUT')

def open1(a):

operation = a.lower()

if operation == 'exit' or operation == 'close':

pass

else:

try:

import os

os.system('start ' + operation)

except:

pass

1. calculator:

from math import \*

def calculator():

print()

print('pi =', pi)

while True:

print()

cal = input('DO YOUR OPERATION DIRECTLY: ').lower()

if cal == 'exit' or cal == 'close':

break

else:

print()

cal2 = eval(cal)

print(cal2)

return cal2

1. count:

def count():

print()

input1 = input('DO YOU WANT TO COUNT THE NUMBER EITHER ASCENDING OR DESCENDING: ').upper()

print()

a1 = int(input('ENTER THE STARTING NUMBER: '))

print()

a2 = int(input('ENTER THE ENDING NUMBER: '))

print()

if input1 == 'ASCENDING' or input1 == 'ASCENDING ORDER' or input1 == '0':

if a1 < a2:

count1 = 0

ts = ''

for i1 in range(a1, a2 + 1):

ts = ts + str(i1) + ', '

print(i1, end=', ')

count1 += 1

if count1 == 5:

print()

count1 = 0

continue

else:

continue

return ts

print()

elif a1 > a2:

count1 = 0

ts = ''

for i1 in range(a2, a1 + 1):

ts = ts + str(i1) + ', '

print(i1, end=',')

count1 += 1

if count1 == 5:

print()

count1 = 0

continue

else:

continue

return ts

print()

elif a1 == a2:

print(a1)

return a1

print()

if input1 == 'DESCENDING' or input1 == 'DESCENDING ORDER' or input1 == '1':

if a1 < a2:

count1 = 0

ts = ''

for i1 in range(a2, a1 - 1, -1):

ts = ts + str(i1) + ', '

print(i1, end=',')

count1 += 1

if count1 == 5:

print()

count1 = 0

else:

continue

return ts

print()

elif a1 > a2:

count1 = 0

ts = ''

for i1 in range(a1, a2 - 1, -1):

ts = ts + str(i1) + ', '

print(i1, end=',')

count1 += 1

if count1 == 5:

print()

count1 = 0

else:

continue

return ts

print()

elif a1 == a2:

print(a1)

return a1

print()

1. hmlv:

def hi():

print()

input2 = open('G:\\My Drive\\FRIDAY\\GENERAL\_QUESTIONS\\TEXT\\hi.txt', 'r')

p = input2.read()

print(p)

return p

def love():

print()

input2 = open('G:\\My Drive\\FRIDAY\\GENERAL\_QUESTIONS\\TEXT\\love.txt', 'r')

p = input2.read()

print(p)

return p

def marry():

print()

input2 = open('G:\\My Drive\\FRIDAY\\GENERAL\_QUESTIONS\\TEXT\\marry.txt', 'r')

p = input2.read()

print(p)

return p

def voice1():

print()

input2 = open('G:\\My Drive\\FRIDAY\\GENERAL\_QUESTIONS\\TEXT\\voice.txt', 'r')

p = input2.read()

print(p)

return p

1. random1:

def random1():

print()

input1 = int(input('''ENTER THE STARTING NUMBER: '''))

input2 = int(input('''ENTER THE ENDING NUMBER: '''))

print()

if input1 > input2:

import random

a2 = random.randint(input2, input1)

print(a2)

elif input1 < input2:

import random

a2 = random.randint(input1, input2)

print(a2)

elif input1 == input2:

a2 = input1

print(a2)

return a2

1. tdmy:

def today():

print()

import datetime

x = datetime.datetime.now()

a = str(x.strftime('%A')) + ' ' + str(x.strftime('%d')) + '-' + str(x.strftime('%B')) + '(' + str(x.strftime('%m')) + ')' + '-' + str(x.strftime('%Y')) + ' ' + str(x.strftime('%I')) + ':' + str(x.strftime('%M')) + ' ' + str(x.strftime('%p'))

print(a)

print()

return a

def time():

print()

import datetime

x = datetime.datetime.now()

a = str(x.strftime('%I')) + ':' + str(x.strftime('%M')) + ' ' + str(x.strftime('%p'))

print(a)

print()

return a

def day():

print()

import datetime

x = datetime.datetime.now()

a = str(x.strftime('%A'))

print(a)

print()

return a

1. weather:

import python\_weather

import asyncio

import os

try:

async def getweather():

async with python\_weather.Client(format=python\_weather.IMPERIAL) as client:

print()

input2 = input('FOR WHICH LOCATION YOU WOULD LIKE TO KNOW THE WEATHER CONDITION: ').lower()

print()

weather = await client.get(input2)

temperature = weather.current.temperature

print("THE TEMPERATURE IN", input2.upper(), "IS: ", str(temperature) + ' F')

print()

for forecast in weather.forecasts:

print("THE PRESENT TIME IN", input2.upper(), "IS: ", forecast.date)

print()

ast = str(forecast.astronomy)

ast = ast.replace('<', '')

ast = ast.replace('>', '')

ast = ast.strip()

ast = ast.split()

ast.pop(0)

ast = tuple(ast)

for i in ast:

i = i.replace('\_', ' ')

i = i.replace('=', ' = ')

i = i.replace('datetime.time(', '')

i = i.replace(')', '')

i = i.replace(',', ' : ')

i = i.upper()

if 'SUN' in i:

print(i, end='')

else:

print(i)

break

for forecast in weather.forecasts:

for hourly in forecast.hourly:

type = str(hourly)

type = type.split()

type.pop(0)

type.pop(0)

type.pop(0)

temp = ''

for i in type:

i = i.replace('=', ' = ')

i = i.replace('️>', '')

i = i.replace("'", '')

i = i.upper()

i = i.strip()

i = i + ' '

temp = temp + i

temp = temp.strip()

print(temp)

break

break

except:

pass

if \_\_name\_\_ == "\_\_main\_\_":

if os.name == "nt":

asyncio.set\_event\_loop\_policy(asyncio.WindowsSelectorEventLoopPolicy())

asyncio.run(getweather())

1. website1:

def website1():

print('ENTER THE NAME OF THE WEBSITE WITH SUFFIX AND PREFIX')

print()

while True:

input1 = input('ENTER THE NAME OF THE WEBSITE: ').lower()

if input1 == 'exit' or input1 == 'close':

break

else:

import webbrowser

webbrowser.register('chrome',

None,

webbrowser.BackgroundBrowser('C:\Program Files\Google\Chrome\Application\chrome.exe'))

webbrowser.get('chrome').open(input1)

def website2(input1):

input1 = input1.lower()

import os

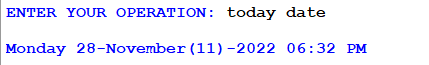
os.system('start chrome ' + input1)

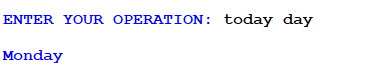
print("WEBSITE IS OPENED IN THE GOOGLE BROWSER")

EXECUTION SCREENSHORT / SAMPLE OUTPUT:

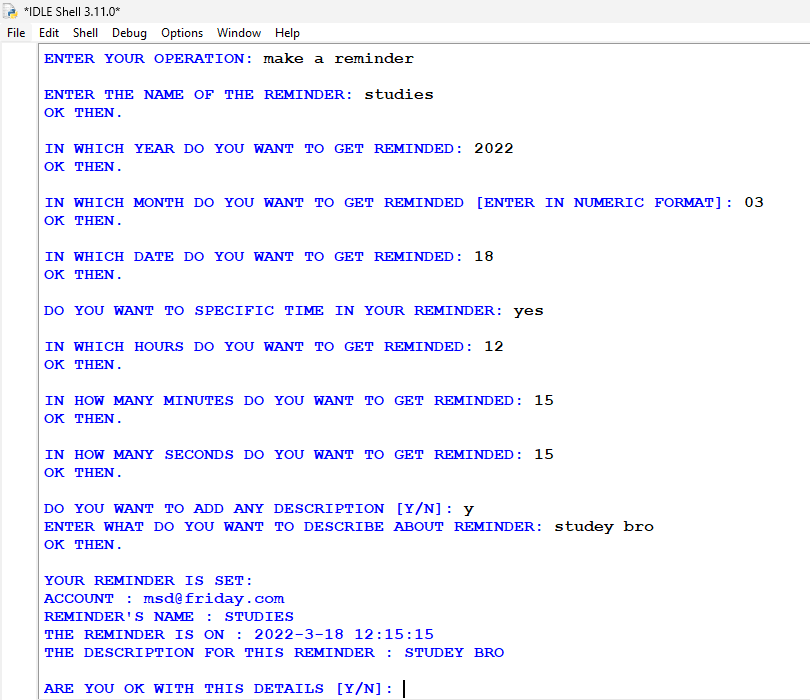




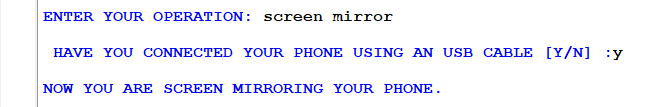


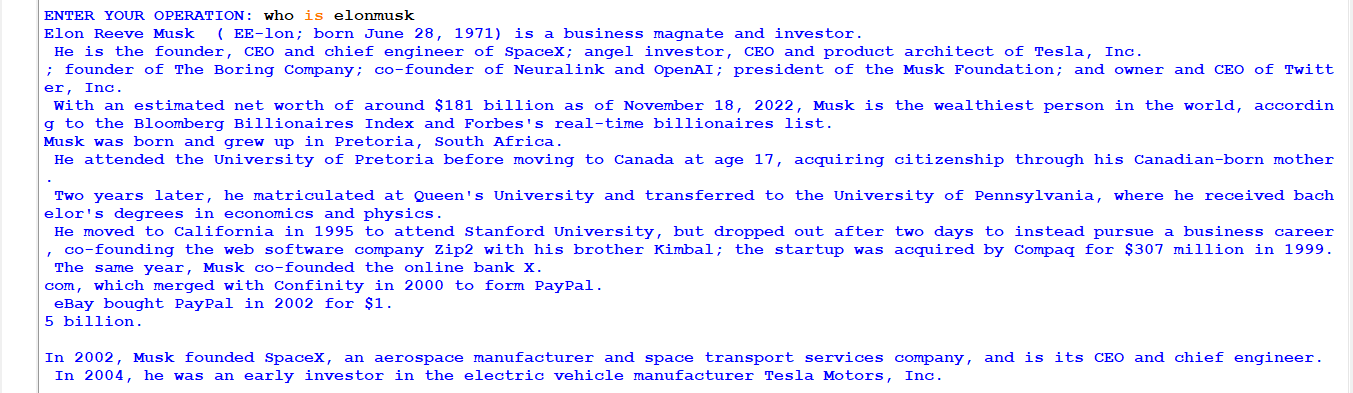




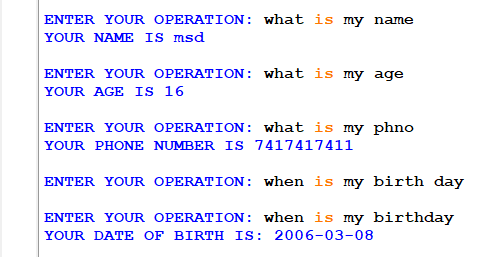


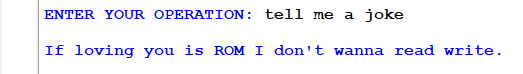


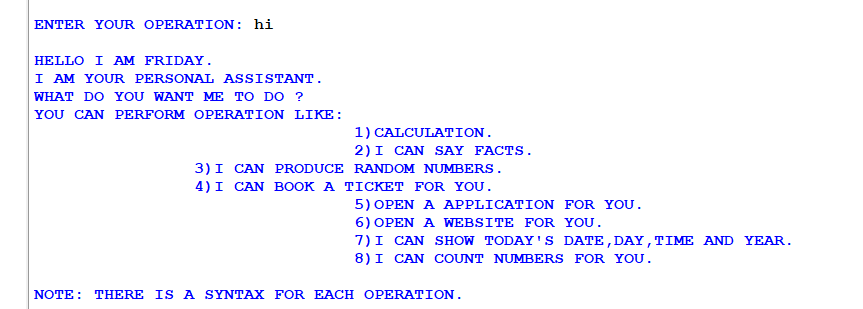


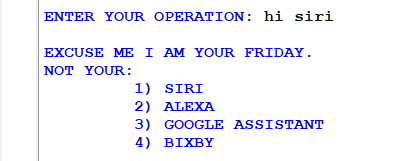


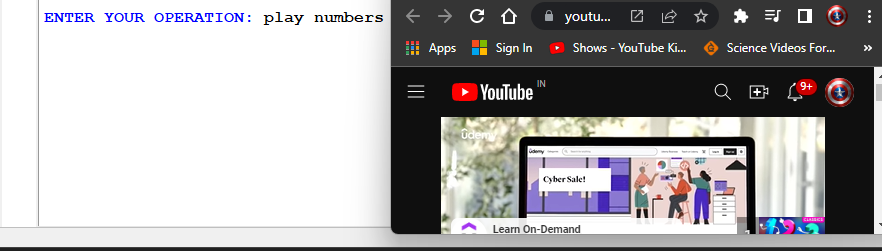












**FUTURE ENHANCEMENTS:**

* Control your devices and your smart home
* Access information from your calendars and other personal information
* Find information online, from restaurant bookings to directions, and news
* Control your music
* Play content on your Chromecast or other compatible devices
* Run timers and reminders
* Make appointments and send messages
* Open apps on your phone
* Read your notifications to you
* Real-time spoken translations
* Play games

Future enhancements listed above are somethings which we were not able to finish in due time and we lacked both in time and knowledge however we have presented our efforts, fully we would like to add these features in future.

Our aim is to make a perfect personal assistant for the users who desire our software and make it very compatible with the user.

**BIBLIOGRAPHY**:

For making this project we have taken references from

* Sumita Arora
* class 12 ncert book
* www.pypi.org
* [www.geeksforgeeks.org](http://www.geeksforgeeks.org)