A project report on

WEB ASSISTANT

Submitted in partial fulfillment of the requirements for the Degree of

B. Tech in Computer Science

by

TUHINA CHAND (1605080)
ANKIT MISHRA (1605101)
ARUNDHATI JHA (1605107)
SHUVAM SUKLABAIDYA(1605156)

under the guidance of

Prof. Amiya Kumar Dash

School of Computer Engineering
Kalinga Institute of Industrial Technology
Deemed to be University
Bhubaneswar

November 2019

KALINGA INSTITUTE
OF INDUSTRIAL TECHNOLOGY
Deemed to be University U/S 3 of the UGC Act, 1956

CERTIFICATE

This is to certify that the project report entitled " **Web Assistant"** submitted by

TUHINA CHAND	**ROLL NO. 1605080**
ANKIT MISHRA	**ROLL NO. 1605101**
ARUNDHATI JHA	**ROLL NO. 1605107**
SHUVAM SUKLABAIDYA	**ROLL NO. 1605156**

in partial fulfillment of the requirements for the award of the **Degree of Bachelor of Technology** in **Discipline of Engineering** is a bonafide record of the work carried out under my(our) guidance and supervision at School of Computer Science and Engineering, Kalinga Institute of Industrial Technology, Deemed to be University.

Prof. Amiya Kumar Dash
Assistant Professor
CSE,KIIT University.

.....

The Project was evaluated by us on 06/12/2019

EXAMINER 1 EXAMINER 2

EXAMINER 3 EXAMINER 4

ACKNOWLEDGEMENTS

We acknowledge the guidance of the supervisors like Prof Amiya Kumar Dash sir, efforts of Group Coordinator, FIC (Project), Program Head and Dean of the School. Besides, we also acknowledge the efforts of the staff , peers and the help from outsiders and agencies who have directly or indirectly supported and contributed in the completion of this project.

TUHINA CHAND
ANKIT MISHRA
ARUNDHATI JHA
SHUVAM SUKLABAIDYA

ABSTRACT

Web Assistant is a digital assistant that uses voice recognition to provide aid to users through speech recognition applications.

The main objective of web assistant is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet.They are used to ease the complicated tasks for the users.

For this project we have used PYTHON as our backend language,web technologies such as HTML,CSS,Bootstrap,JavaScript as our frontend,for our database we have used MySQL database using PHPMyAdmin and for connecting both frontend and backend we have used Flask RESTful api.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

WEB-ASSISTANT - a digital assistant.

The basic idea behind this project is to create a simple stand-alone application that helps less tech savvy people in the world to use the computer without feeling ignorant or computer illiterate.

The main objective of WEB-ASSISTANT is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet. They are used to ease the complicated tasks for the users. For this project we have used PYTHON as our back-end language,web technologies such as HTML,CSS,Bootstrap,JavaScript as our front-end,for our database we have used MySQL database and for connecting both front-end and back-end we have used Flask Restful api.

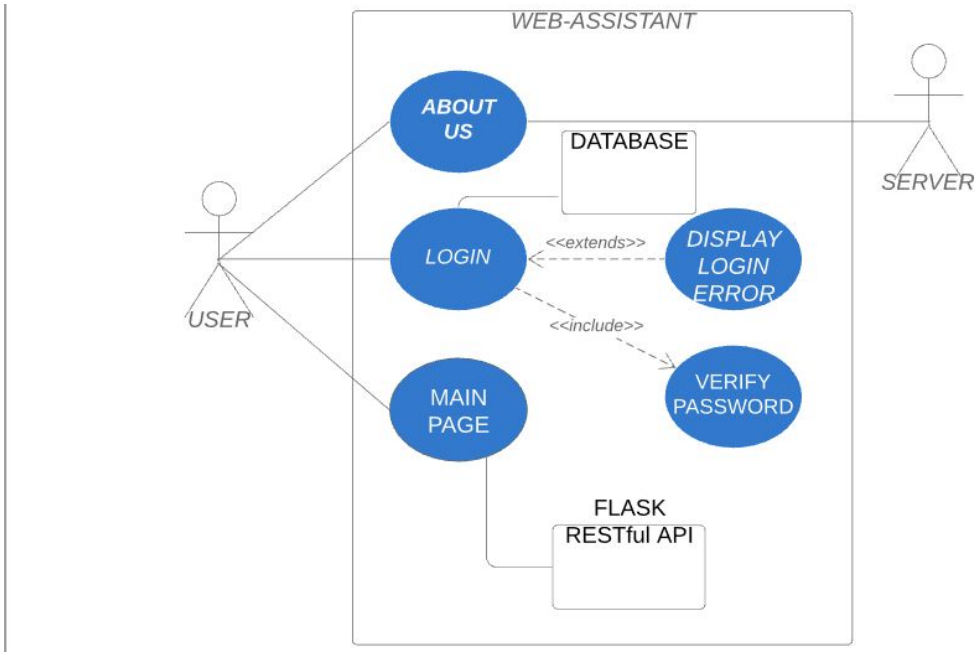
Computers have become a very important devices and as well as less expensive over time. The application works like Siri/ Google Assistant etc. But the application deals with the computer itself mainly. The U.I of the application is self-explanatory and minimal.Currently it takes text as input as most of the people are not very good at speaking. The application was built on and for a Windows 10 OS thus restricting it to the Windows 10 OS alone. WEB-ASSISTANT is compatible with all versions of the Windows 10 OS.

The system also assumes that the user has minimal English knowledge as of now. WEB-ASSISTANT was built to help people with limited computer knowledge but it is also important to note that the other class of users might find some specific functionalities such as system logging useful.The people who are not familiar with getting around on their own on a computer can use this application as it abstracts away all the steps and presents only the most important. Like they can just say to open an application, file? Want to save a file? Want to play a video? Want to calculate a mathematical problems? Just say it as you would to a person, it will do the work for you. It can also help with getting

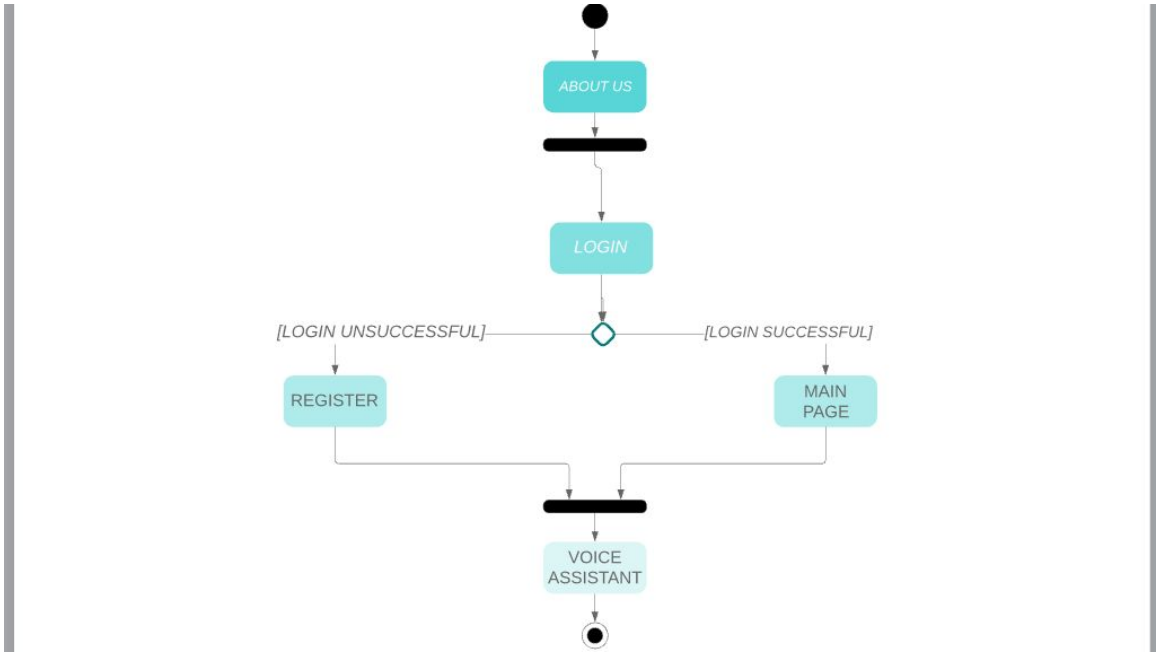
around web by opening, closing, etc a web page just by saying it to do so. It is the solution for the parents who got a computer from their son/daughter and can't operate it to talk to them. It is the solution for the people who are willing to make a computer as their real assistant.

CHAPTER 2: PROJECT IMPLEMENTATION

- **Block Diagrams**
  - USE-CASE DIAGRAM



  - ACTIVITY DIAGRAM



- **Hardware Used**
  - OS:Microsoft Windows 10
- **Software Used**
  - Software Used:Google Chrome,MySQL,PHPMyAdmin,Brackets,SublimeText, Adobe Photoshop
  - Languages Used:Python,PHP,HTML,CSS,JavaScript,BootStrap,Flask RESTful api
- **Implementation**
  - **Frontend**

    Front-end is a term that involves the building of web pages and user interfaces for web-applications. It implements the structure, design, behavior, and animation of everything you see on the screen when you open up websites, web applications, or mobile apps. The core 3 technologies that all modern front-end web developers work to master are HTML5, CSS, and JavaScript.

    Html5-: HyperText Markup Language, the document standard of the World Wide Web for creating web pages and web applications. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

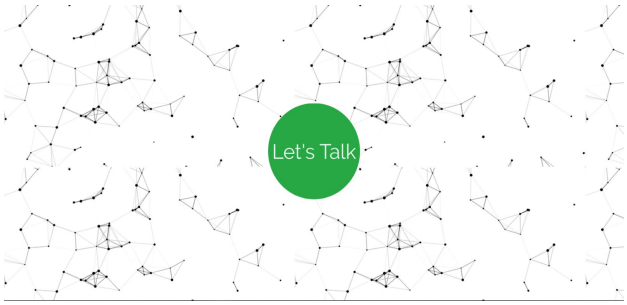
    Css-: A stylesheet language that is used for presentation and formatting content on the web-pages, including font, size, color, spacing, border and location of HTML information.

    Javascript-: A client-side scripting language primarily used to make web pages interactive. In other words, it's responsible for the 'behavior' of a website, i.e. how HTML elements and CSS style animate and move around on the page.

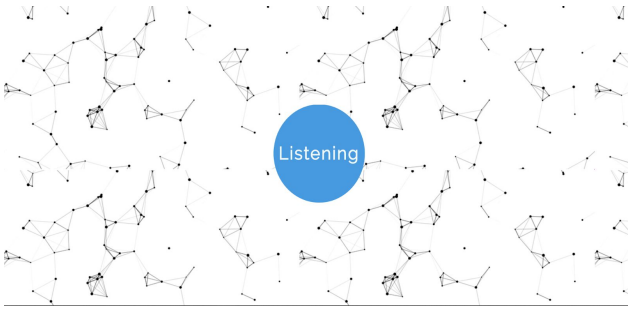
    Bootstrap-: An open source toolkit for building responsive, mobile-first projects on the web with HTML, CSS, and JavaScript. Enables to quickly prototype ideas or build the

entire app using Sass variables and mixins, responsive grid system, extensive pre built components, and powerful plugins built on jQuery.

The frontend implemented after signing in:



It is where the user will interact with the web assistant and let the assistant search for anything for the user like searching for some movies, playing songs online, play youtube videos, can also be used to perform mathematical calculations and also can be used to save data that the user wants to save like a reminder.



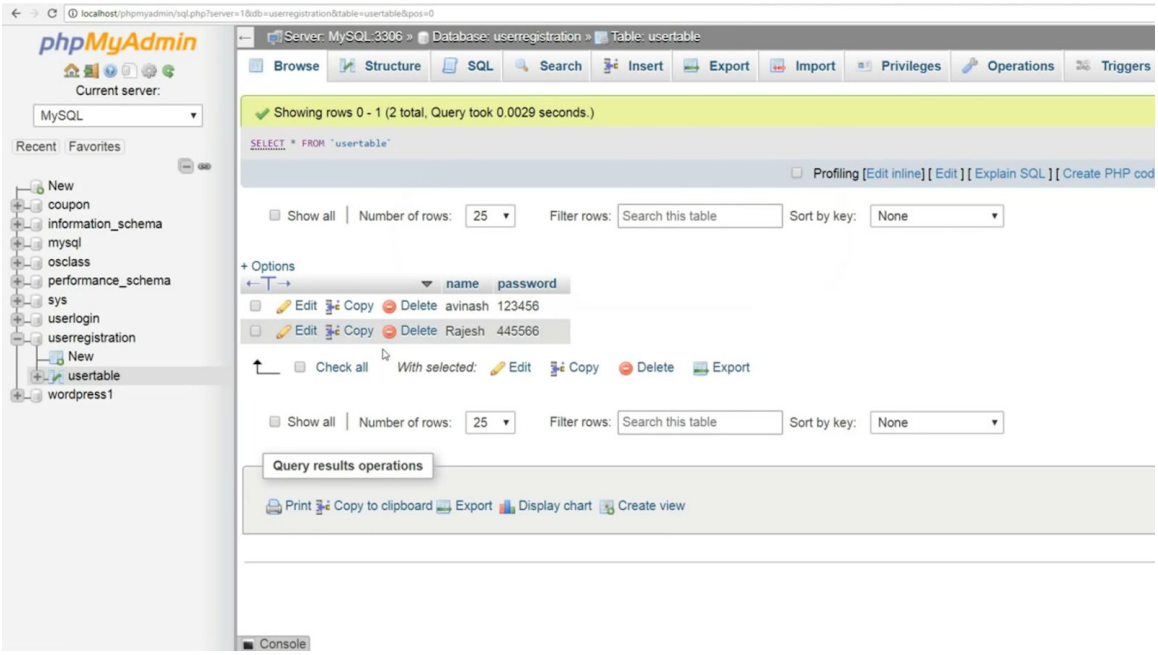
The web assistant here will listen and send the audio requested by the user to the flask RESTful api and wait for the response from the api.

After getting the result it will render accordingly.

- **Database**

  PHP WITH **MySQL Database:** In order to authenticate the use of the voice assistant in the website, we have

created a login-logout system where a MySQL database in the PHP pages with phpMyAdmin is storing the user details. We check the login credentials before permitting use of web assistant and also provide registration option to new users.



- **Flask RESTful api**

  Flask RESTful is an extension for Flask that adds support for quickly building REST APIs. It is a lightweight abstraction that works with your existing libraries. Flask RESTful encourages best practices with minimal setup. If you are familiar with Flask, Flask RESTful should be easy to pick up.

- **Backend**
  - **Package Used:**
    - **Speech Recognition:**It converts the spoken words into text.
    - **WebBrowser:**It provides a high level interface to allow web-based documents to users.
    - **Wolfram Alpha:**It is a highly complicated knowledge engine.Using this we can answer highly complex mathematical operations,current affairs,weather forecast etc.

- **Wikipedia:**It is a library that makes it easy to access and parse data from wikipedia.
- **OS:**It provides a way of using OS dependent functionality.
- **Pyperclip:**It provides cross-platform python module for copying and pasting text in clipboard.
- **Win32com.client:**It is used to convert the text into speech.

■ **Coding Implementation:**

- To convert voice into text:

Import package:

```
import speech_recognition as sr
```

Call Recognizer method:

```
r = sr.Recognizer()
r.pause_threshold = 0.7
r.energy_threshold = 400
```

pause_threshold and energy_threshold (loudness) are attributes for the speech.

Use Microphone method :

```
with sr.Microphone() as source:

    try:
        print("Please Speak")
        audio = r.listen(source, timeout = None)
        message = str(r.recognize_google(audio))
        print('You said: ' + message)
```

We are using a microphone as our source for speech.

recognize_google() is a google api for converting voice into text message.

- To convert text into speech:

```
v =wincl.Dispatch("SAPI.SpVoice")
```

```
shell = wincl.Dispatch("WScript.Shell")
v.Speak('Hello! For a list of commands, plese say "keyword list"...')
```

SAPI.SpVoice is an object that brings the text-to-speech capabilities into the application using SAPI automation.

Speak() method is used to deliver the text into speech form.

- Connect it to Wolfram alpha:

```
cl = wolframalpha.Client('JRGPQ3-G2GUPJ7UGU')
```

It will connect the wolfram alpha knowledge engine with the application.

- Define specific keywords:

```
keywd = 'keyword list'
google = 'search for'
acad = 'academic search'
sc = 'deep search'
wkp = 'wiki page for'
rdds = 'read this text'
sav = 'save this text'
vid = 'video for'
wtis = 'what is'
wtar = 'what are'
whis = 'who is'
whws = 'who was'
when = 'when'
where = 'where'
how = 'how'
paint = 'paint'
lsp = 'silence please'
lsc = 'resume listening'
stoplst = 'stop listening'
```

- Perform various tasks:

Google Search:

```
if google in message:

    words = message.split()
    del words[0:2]
    st = ' '.join(words)
    print('Google Results for: '+str(st))
    url='http://google.com/search?q='+st
    webbrowser.open(url)
    v.Speak('Google Results for: '+str(st))
```

webbrowser.open() is used to open the url in the google chrome.

Wikipedia search:

```
elif wkp in message:

    try:

        words = message.split()
        del words[0:3]
        st = ' '.join(words)
        wkpres = wikipedia.summary(st, sentences=2)

        try:

            print('\n' + str(wkpres) + '\n')
            v.Speak(wkpres)

        except UnicodeEncodeError:
            v.Speak(wkpres)

    except wikipedia.exceptions.DisambiguationError as e:
        print (e.options)
        v.Speak("Too many results for this keyword. Please be more specific and try again")
        continue

    except wikipedia.exceptions.PageError as e:
        print('The page does not exist')
        v.Speak('The page does not exist')
        continue
```

wikipedia.summary() method extracts the information from wikipedia.

Exceptions:

If encoding is unknown then UnicodeEncodeError

If ambiguation in message or result the wikipedia.exceptions.DisambiguationError.

If page is not found then wikipedia.exceptions.PageError.

Save the Text :

```
elif sav in message:
    with open(r'D:\Files\files\python\srpyper.txt', 'a') as f:
        words=message.split()
        del words[0:3]
        st=' '.join(words)
        pyperclip.copy(st)
        f.write(pyperclip.paste())
    print("Saving your text to file")
    v.Speak("Saving your text to file")
```

pyperclip.copy() and pyperclip.paste() are copy and paste functions for clipboard.

Search based on knowledge engine:

```
elif wtis in message:

    try:

        scq = cl.query(message)
        sca = next(scq.results).text
        print('The answer is: '+str(sca))
        v.Speak('The answer is: '+str(sca))

    except UnicodeEncodeError:

        v.Speak('The answer is: '+str(sca))

    except StopIteration:

        words = message.split()
        del words[0:2]
        st = ' '.join(words)
        print('Google Results for: '+str(st))
        url='http://google.com/search?q='+st
        webbrowser.open(url)
        v.Speak('Google Results for: '+str(st))
```

next() is an iterator provided by wolfram alpha that shows the next result instead of full result and query() is used to send the query to wolfram alpha engine.

Open any system application:

```
elif paint in message:
    os.system('mspaint')
```

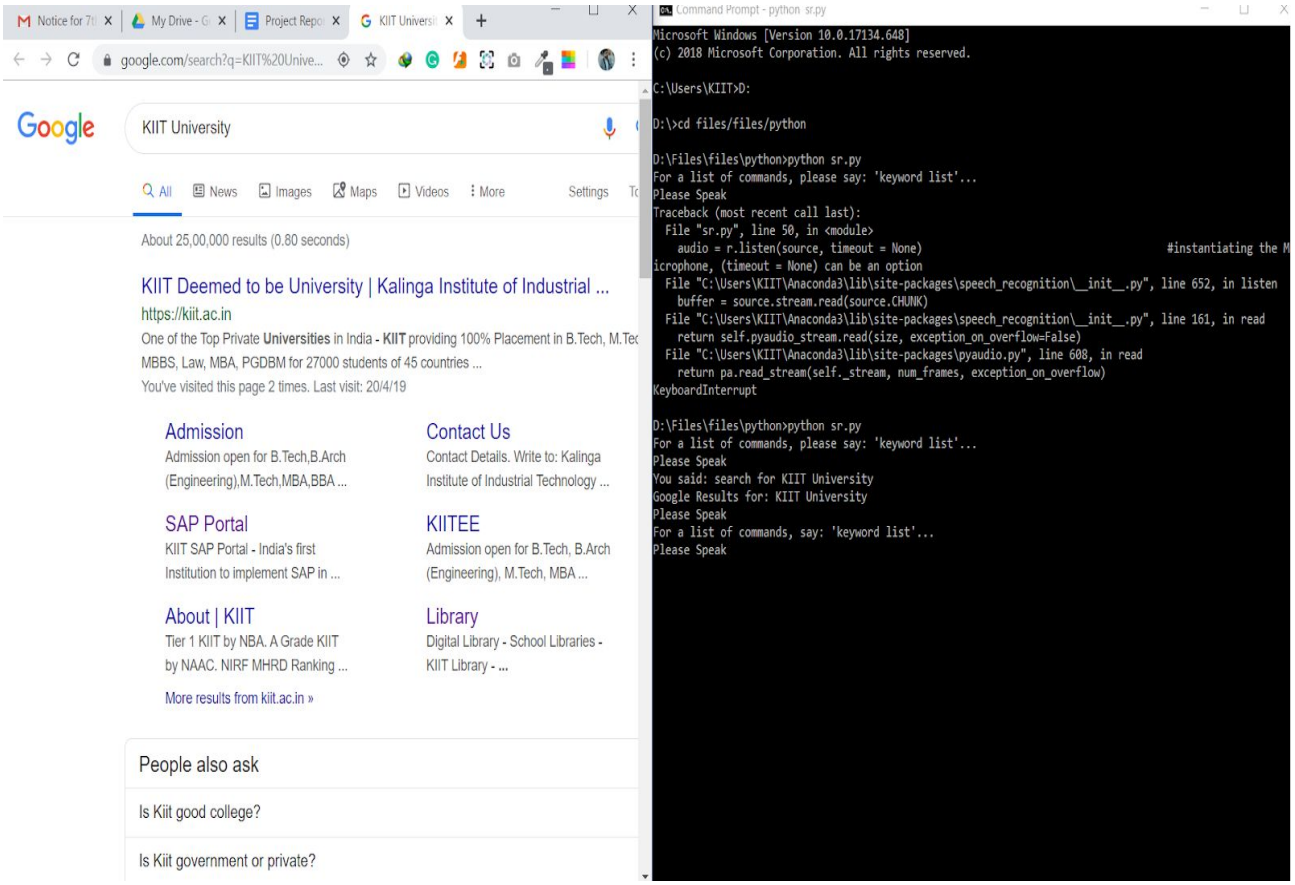
os.system() directly opens the application that is specified.

Pause Listening:

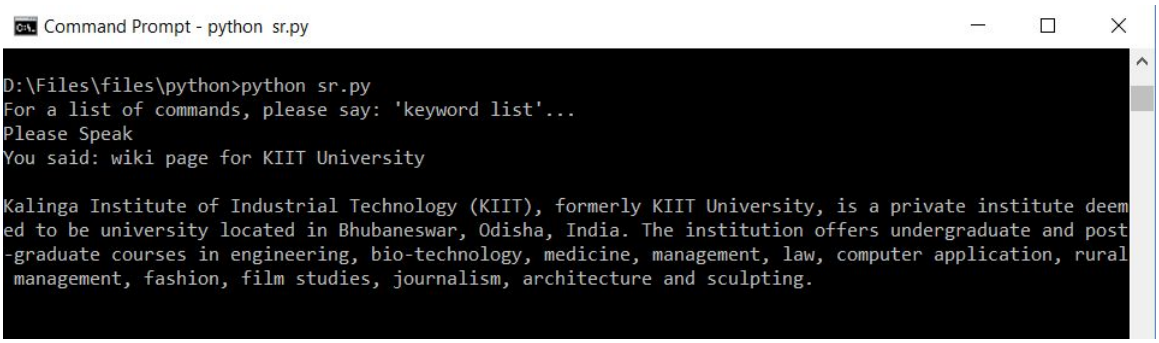
```
elif lsp in message:

    v.Speak('Listening is paused')
    print('Listening is paused')
    r2 = sr.Recognizer()
    r2.pause_threshold = 0.7
    r2.energy_threshold = 400

    while True:

        with sr.Microphone() as source2:

            try:

                audio2 = r2.listen(source2, timeout = None)
                message2 = str(r.recognize_google(audio2))

                if lsc in message2:
                    v.Speak('I am listening')
                    break

                else:
                    continue

            except sr.UnknownValueError:
                print("Listening is paused. Say resume listening when you're ready...")

            except sr.RequestError:
                v.Speak("I'm sorry, I couldn't reach google")
                print("I'm sorry, I couldn't reach google")
```

CHAPTER 3: RESULTS & DISCUSSIONS

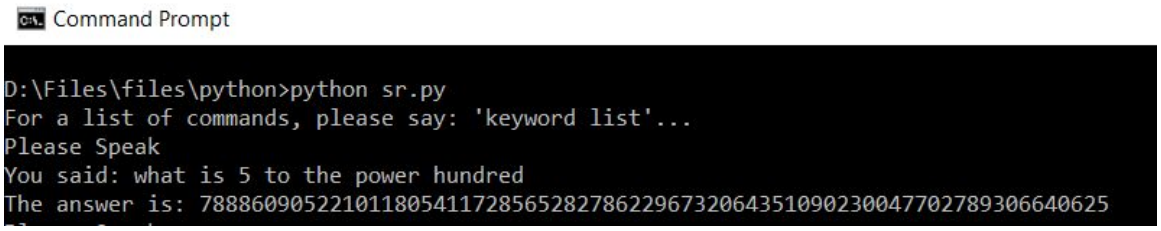
- **Google search:**



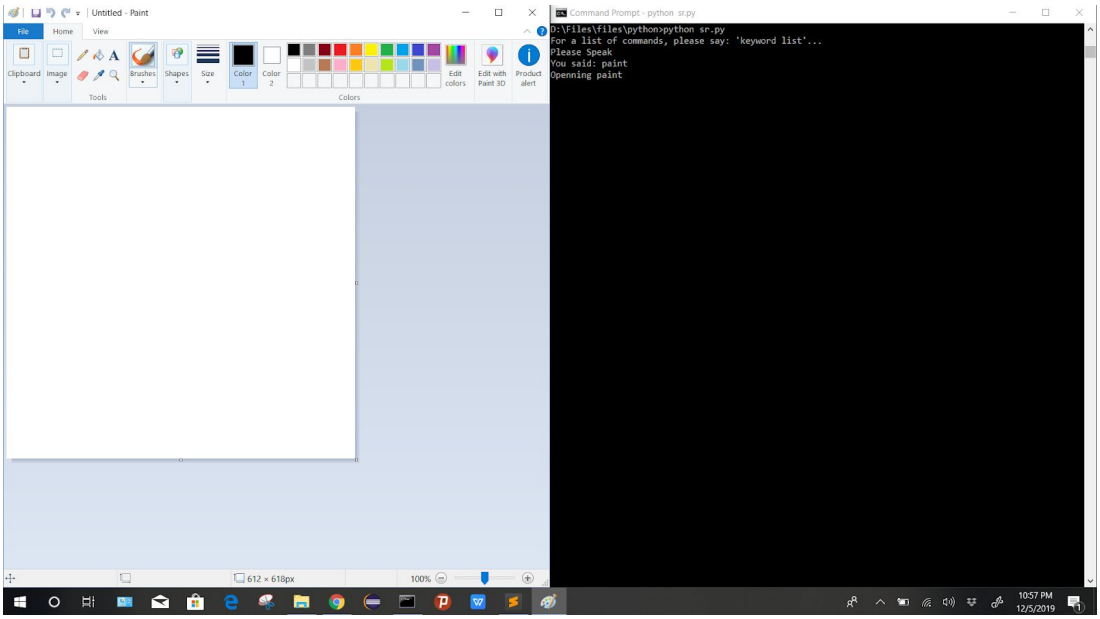
- **Wikipedia search:**



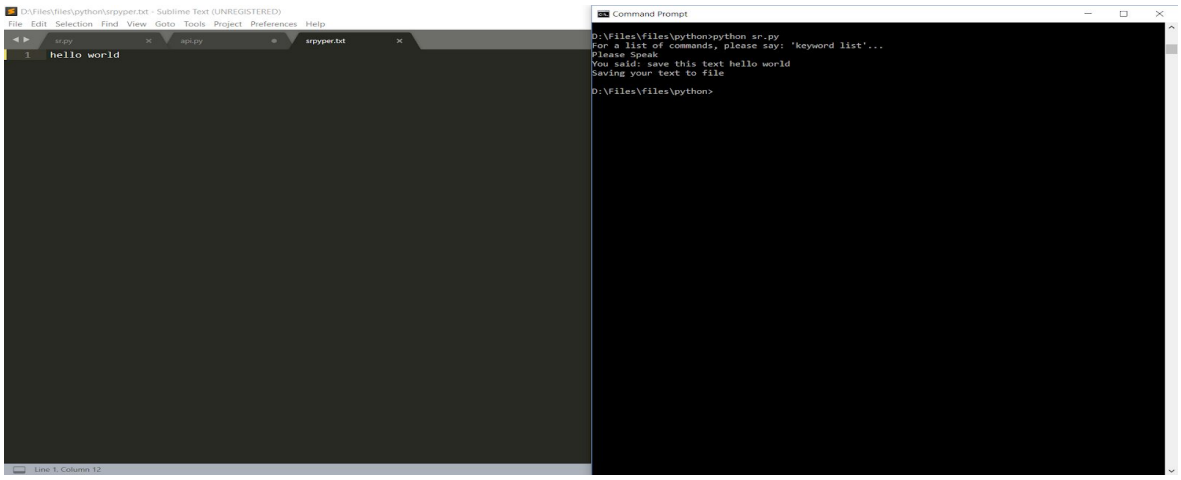
- **Mathematical Operations:**



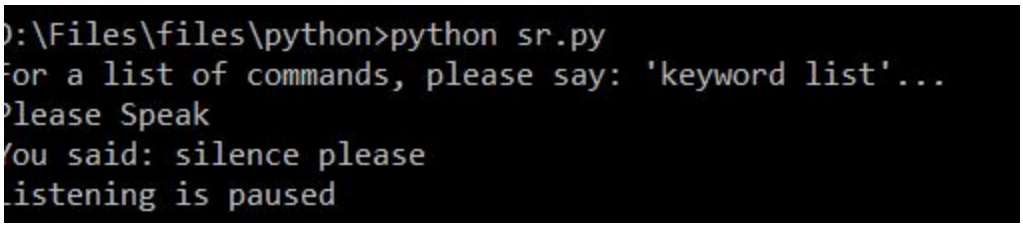
- **Open Paint:**



- **Saving the text:**



- **Pause Listening:**



Similar to this tasks it also performs-deep search,academic search,read the text,save the text and answers to the questions such as what is,what are,who is, who was,when,where,how.

It can also open the OS applications such as paint.

We can pause listening or resume listening or stop the application just by giving speech commands.

All the various exceptions has been handled some of them are-PageError,UnicodeEncodeError,DisambiguationError,StopIteration,UnknownValueError,RequestError etc.

CHAPTER 4: CONCLUSION AND FUTURE WORK

4.1 CONCLUSION

WEB-ASSISTANT has been very useful since it aid users to perform various web related tasks and it is user friendly thereby helping the users to perform the tasks just by listening to them. With the help of WEB-ASSISTANT we can perform various tasks like deep searching, play music, search for videos ,perform calculations and many more tasks thereby helping the user.

The WEB-ASSISTANT has a great future ahead and it will be implemented to almost all the websites in the future thereby spreading awareness and benefiting the user by keeping them at ease.

4.2 FUTURE WORK

At present, we have completed the frontend and the backend. The connection between the frontend and the backend using flask RESTful api and connection between sign-in and login page with the MySQL database are left for the future due to lack of time and knowledge. So the next step for us will be to establish a connection between the frontend and the backend and also establish a connection between sign-in and login page with the MySQL database.

4.3 EFFECTS

Digital assistants embody the dream of an effortless future, free from the shackles of yesteryear: a tool which caters to users' needs, excels at anticipating their wants, and delivers a personalized online environment. While digital assistants can certainly offer great value, a closer look reveals how in an algorithm and data driven world a dominant digital assistant may ultimately serve the interests of corporations rather than consumers. Such assistants may be used to establish a controlled and manipulated personalized environment in which competition, welfare, privacy, and democracy give way to corporate interests. The future is not necessarily bleak, but requires our attention if users want the leading assistants to match the effortless dream.

REFERENCES

From books

1. Learning Python,David Ascher and Mark Lutz(2003),1005 Gravenstein Highway North,Sebastopol,CA 95472.Published by O'Reilly Media,Inc.

2. HTML & CSS: Design and Build Websites, Jon Duckett(2012).

From web pages

1. https://www.geeksforgeeks.org/speech-recognition-in-python-using-google-speech-api/
2. http://timgolden.me.uk/pywin32-docs/html/com/win32com/HTML/QuickStartClientCom.html
3. https://www.w3schools.com/

INDIVIDUAL CONTRIBUTION REPORT:

WEB ASSISTANT

ANKIT MISHRA

1605101

Abstract:Web Assistant is a digital assistant that uses voice recognition to provide aid to users through speech recognition applications.The main objective of web assistant is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet.They are used to ease the complicated tasks for the users.

Contribution and findings: After finalizing the project ,I started gathering information about technologies required to implement all the functionalities required for the backend.After gathering information ,I started learning about the new technologies that are required for the project that includes implementation of speech recognition ,flask RESTful api and win32com.client api.Flask RESTful api will be used for connecting frontend with backend.After that ,I started writing code.While writing code and testing for each functionality,I came across various exceptions and errors such as system application was not opening,Valueerror,requesterror etc.After resolving all the errors and exceptions, I started testing for each task.

Contribution to project report preparation: For the project report,I contributed in abstract section,in project implementation section I contributed in backend implementation,in results & discussions section,in planning and project management section and in reference section.

Contribution for project presentation and demonstration:I contributed in backend and flask api section for project presentation and demonstration.

Full Signature of Supervisor: Full signature of the student:

....

WEB ASSISTANT

SHUVAM SUKLABAIDYA

1605156

Abstract:Web Assistant is a digital assistant that uses voice recognition to provide aid to users through speech recognition applications.The main objective of web assistant is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet.They are used to ease the complicated tasks for the users.

Contribution and findings: In this project, I have contributed as a Front End Developer who creates and codes the visual front-end elements of a website, software or applications. First of all i have used these web technologies: HTML , CSS, Javascript to help build web pages such as after signing in we will be logged in to a page which will directly interact with the Back End. For building the web pages I have used the above web technologies and also help in designing the background along with some editing of logos in Photoshop. Also , along with some knowledge from many different websites I have come with some ideas which really helped me in web designing. Our main focus for every Front End Developer is to improve the user experience by evaluating how much the user is satisfied.

Contribution to project report preparation: In report preparation, I have contributed myself in writing about the front end after signing in, conclusion of web-assistant and the future work along with the references.

Contribution for project presentation and demonstration: I contributed in front end section, conclusions, future work and references.

Full Signature of Supervisor: Full signature of the student:

.....
.....

WEB-ASSISTANT

TUHINA CHAND

1605080

Abstract: Web Assistant is a digital assistant that uses voice recognition to provide aid to users through speech recognition applications.The main objective of web assistant is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet.They are used to ease the complicated tasks for the users.

Contributions and findings: Front-end Developers are responsible for implementing visual elements that users see and interact within a web application. I basically focused on building the UI (User interfaces) for the users of our website or web application. I consider a few things: Accessibility, Aesthetics, Performance, Security, Quality of code. The aim of the project is making software interactions simple, fun, easy and effective for that I used different web technologies such as html5, Css, javascript and bootstrap and some inbuilt templates. During this project I went through lots of websites to find out the idea for web designing. I need good understand of image authoring tools, to be able to crop, resize, or perform small adjustments on an image. Familiarity with tools such as Photoshop is a plus. Also need a basic understanding of javascript and bootstrap to work on inbuilt library to make the web page responsive. The website contains different sections such as about us, team, form, contact and footer which is design with the help of css sheet. I mainly work on about us, team and footer section. Also responsible for writing content using Plain Language and search optimization techniques and developed content style guides. And focused on evaluating the quality of a user's experience when interacting with a system. Tests, analyzes and reports results, and makes recommendations to improve effectiveness, efficiency, and overall satisfaction of the user.

Contribution to project report preparation: I contributed in introduction part and the effect of this website on societies and environment.

Contribution for project presentation and demonstration: I contributed in front end section for project presentation and demonstration.

Full Signature of Supervisor: Full signature of the student:

……. ……..

WEB ASSISTANT

ARUNDHATI JHA

1605107

Abstract : Web Assistant is a digital assistant that uses voice recognition to provide aid to users through speech recognition applications.The main objective of web assistant is to aid users to perform various web related tasks such as searching for some content over the internet or performing calculations specified by user or download files from the internet.They are used to ease the complicated tasks for the users.

Contributions and findings : After the finalization of project topic, I tried to gather information about providing authentication to the website and database that it is using. I used MySQL Database for storing user details like username,password etc. I created a table in SQL database in the "phpMyAdmin" and inserted columns like user(primary), password in the table. This database helps us in authenticating a user's entry into the website voice assistant as well as dynamically stores the credentials of the new registrations. I used php web pages for scripting where I connected the SQL database and checked for correct login credentials for an already registered user whereas it can also be used to create new user credentials in the SQL database in case of new registrations. While registering a user, I also checked if a user already exists with the same username and in that case did not allow new registration with that username. While logging in, if the username and password are correct , I redirected the website to the page where the voice assistant works.

Contribution to project report preparation: For the project report, I contributed in the certification and acknowledgement section, table of contents, block diagrams(Use-Case diagram and Activity Diagram) and the front-end database(PHP and MySQL) in the project implementation section.

Contribution for project presentation and demonstration: I contributed in front end implementation section using database MySQL and PHP.

Full Signature of Supervisor: Full signature of the student:

……. ……..