A

Minor Project Report

on

**ELISA-A Virtual Assistant**

Submitted in partial fulfillment of the requirements for the award of degree of

**BACHELOR OF TECHNOLOGY**

by

**SHWETA MEHRA**

**(IT-4052-2K16)**

**VINIT RATWAYA**

**(IT\_4060-2K16)**

Under supervision of

**MS.MONIKA GUPTA**



**Department of Computer Engineering**

**J. C. BOSE UNIVERSITY OF SCIENCE & TECHNOLOGY, YMCA**

**FARIDABAD-121006**

**May 2019**

**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in this project report titled **ELISA-A VIRTUAL ASSISTANT** in fulfillment of the requirement for the degree of Bachelor of Technology and submitted to “**J. C. Bose** **University of Science and Technology, YMCA, Faridabad**”*,* is an authentic record of my own work carried out under the supervision of MS.MONIKA GUPTA

The work contained in this report has not been submitted to any other University or Institute for the award of any other degree or diploma by me.

**SHWETA MEHRA**

**(IT-4052-2K16)**

**VINIT RATWAYA**

**(IT-4060-2K16)**

**CERTIFICATE**

This is to certify that the project report titled **“ELISA-A Virtual Assistant”** submitted by **Shweta Mehra and Vinit Ratwaya** to “**J. C. Bose** **University of Science and Technology, YMCA, Faridabad**” for the award of the degree of Bachelor of Technology is a record of bonafide work carried out by them under my supervision. In my opinion, the work has reached the standards of fulfilling the requirements of the regulations to the degree

Ms. Monika Gupta

(Supervisor)

Assistant Professor,

Department of Computer Engg

J. C. Bose University of Science and Technology, YMCA, Faridabad

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Content** | **Page No.** |
| **1.** | **INTRODUCTION** | **5** |
| **2.** | **PROBLEM STATEMENT** | **6** |
| **3.** | **MOTIVATION** | **6** |
| **4.** | **OBJECTIVES & SCOPE** | **7** |
| **5.** | **LITERATURE REVIEW** | **8** |
| **6.** | **PROPOSED METHODOLOGY(USE CASE DIAGRAM, DATAFLOW DIAGRAM, FLOWCHART)** | **15** |
| **7.** | **HARDWARE AND SOFTWARE REQUIREMENTS** | **16** |
| **8.** | **RESULTS (SCREENSHOTS, GRAPHS Etc.)** | **18** |
| **9.** | **CONCLUSION** | **25** |

1. **INTRODUCTION**

Elisa is a personal virtual assistant working on various operating system .The assistant uses voice queries and a natural-language user interface to answer questions, make recommendations, and perform actions by delegating requests to a set of Internet services. It can access via English.

It can set reminders ,handling device, send messages via various social networking, weather information service, recognize natural voice without the requirement for keyboard input, and answer questions using information from the Google search engine .The speech recognition system makes use of sophisticated machine learning techniques using python.

It's basically voice control that talks back to you, that understands relationships and context, handle basic sequential inference. Ask Elisa questions, or ask Elisa to do things for you, just like you would ask a real assistant, and Elisa will help keep you connected, informed, in the right place, and on time. You can even use Elisa built-in dictation feature to enter text almost everywhere by simply using your voice.

Elisa catches the sound of your speech and instantly converts it into a digital form that has stored the relevant information. That signal from your laptop will be conveyed wirelessly to a nearby cell tower and through a series of the land line back to your internet service provider where it will communicated with a server, there an algorithm uses machine learning and natural language processing to understand your language and your question.

**2 .PROBLEM STATEMENT**

Many functions are provided by different Google search engines and other assistants .But in the modern era everything is digital service everywhere. As we stand on the brink of the third wave of computing, the concept of a virtual assistant (a digital service looking after a range of our needs) is fast becoming a reality. Other assistants do not provide access to Facebook. This is a major limitation.

**3. MOTIVATION**

We set out with consumers and experts to design a blueprint for the virtual assistant of the future, designing what it will do, how we will interact with it and where its boundaries will lie. This report outlines our findings and five key conclusions:

1. Consumers are looking for a virtual assistant service capable of being accessed from a number of devices. While smartphones will dominate usage in the short term, in the longer term we can expect virtual assistants to be accessed through smart watches, wearable, cars and other connected devices.

2. Consumers expect a modular service, capable of bolting on additional specialist functionality to the core virtual assistant to fulfill specific needs.

3. There’s openness to an ad supported VA business model, with ‘paid recommendations’ offered to the user alongside ‘organic recommendations’.

4. The VA has the potential to be our next gateway to the internet, delivering services that span every aspect of our lives. The research identifies five core service areas: Personal; Community; Economy; Crowd and Environment

**4**. **OBJECTIVES** **AND** **SCOPE**

* ELISA is created to be a truly personal digital assistant – available on the go, no matter where or when we need help, and on whatever device we carry.
* ELISA is the personal digital assistant to help, complete tasks across multiple devices, from PC to phone and vice versa.
* ELISA can understand you like your personal assistant virtually and with

More accurate understanding of your language!

“Remind me to tell my bro about my new website”

Or

“Call my dad”

**5. LITERATURE REVIEW**

‘ELISA-A virtual assistant’, is a platform assistant that uses voice queries and a natural-language user interface to answer questions, make recommendations, and perform actions by delegating requests to a set of Internet services. It can access via English.

It can set reminders, handling device, send messages via various social networking, weather information service, recognize natural voice without the requirement for keyboard input, and answer questions using information from the Google search engine. The user can make queries and Elisa can respond to such queries providing flexibility and reliability to many users.

**TECHNOLOGY STACK**

The following technologies have been used for successful implementation of the app ‘ELISA-A Virtual assistant’.

* Scikit
* Python
* Octave
* MATLAB
* Pycharm

**5.1 SciKit**

Scikits (short for SciPy Toolkits), are add-on packages for SciPy, hosted and developed separately and independently from the main SciPy distribution. All SciKits are licensed under OSI-approved licenses.

Packages are packaged as toolkits (instead of in the main, monolithic SciPy distribution) when:

* The package is deemed too specialized to live in SciPy itself or
* The package has a GPL (or similar) license which is incompatible with SciPy’s BSD license or
* The package is meant to be included in SciPy , but development is still in progress
* [Scikit Learn](https://scikit-learn.org/stable/index.html) is the gold standard. Scikit-learn provides a wide selection of supervised and unsupervised learning algorithms. Best of all, it’s by far the easiest and cleanest ML library.
* Scikit learn was created with a software engineering mindset. It’s a core API design revolves around being easy to use, yet powerful, and still maintaining flexibility for research endeavors. This robustness makes it perfect for use in any end-to-end ML project, from the research phase right down to production deployments.

Scikit Learn is built on top of several common data and math Python libraries. Such a design makes it super easy to integrate between them all. You can pass numpy arrays and pandas data frames directly to the ML algoirthms of Scikit! It uses the following libraries:

* [**NumPy**](http://www.numpy.org/): For any work with matrices, especially math operations
* [**SciPy**](https://www.scipy.org/) : Scientific and technical computing
* [**Matplotlib**](https://matplotlib.org/): Data visualisation
* [**IPython**](https://ipython.org/): Interactive console for Python
* [**Sympy**](https://www.sympy.org/en/index.html): Symbolic mathematics
* [**Pandas**](https://pandas.pydata.org/): Data handling, manipulation, and analysis

Scikit Learn is focused on Machine Learning e.g data modelling. It is not concerned with the loading, handling, manipulating, and visualizing of data. Thus, it is natural and common practice to use the above libraries, especially NumPy, for those extra steps; they are made for each other!

Scikit’s robust set of algorithm offerings includes:

* **Regression:** Fitting linear and non-linear models
* **Clustering:** Unsupervised classification
* **Decision Trees:** Tree induction and pruning for both classification and regression tasks
* **Neural Networks:** End-to-end training for both classification and regression. Layers can be easily defined in a tuple
* **SVMs:** for learning decision boundaries
* **Naive Bayes**: Direct probabilistic modelling

Even beyond that, it has some very convenient and advanced functions not commonly offered by other libraries:

* **Ensemble Methods:** Boosting, Bagging, Random Forest, Model voting and averaging
* **Feature Manipulation**: Dimensionality reduction, feature selection, feature analysis
* **Outlier Detection:** For detecting outliers and rejecting noise
* **Model selection and validation:** Cross-validation, Hyperparamter tuning, and metrics



Figure 1(SciKit)

**5.2 PYTHON**

## What is Python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

* web development (server-side),
* software development,
* mathematics,
* system scripting.

### What can Python do?

* Python can be used on a server to create web applications.
* Python can be used alongside software to create workflows.
* Python can connect to database systems. It can also read and modify files.
* Python can be used to handle big data and perform complex mathematics.
* Python can be used for rapid prototyping, or for production-ready software development.

### Why Python?

* Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.
* Python has a simple syntax similar to the English language.
* Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
* Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
* Python can be treated in a procedural way, an object-orientated way or a functional way.

### Good to know

* The most recent major version of Python is Python 3, which we shall be using in this project. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
* In this l Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans or Eclipse which are particularly useful when managing larger collections of Python files.

### Python Syntax compared to other programming languages

* Python was designed for readability, and has some similarities to the English language with influence from mathematics.
* Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.
* Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

**5.3 OCTAVE**

* Scientific Programming Language
* Powerful mathematics-oriented syntax with built-in plotting and visualization tools
* Free software, runs on GNU/Linux, macOS , BSD, and Windows
* Drop-in compatible with many Matlab scripts

**5.4 MATLAB**

MATLABcombines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly. It includes the Live Editor for creating scripts that combine code, output, and formatted text in an executable notebook.

**5.5 Pycharm**

PyCharm is an [integrated development environment](https://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) used in [computer programming](https://en.wikipedia.org/wiki/Computer_programming), specifically for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) language. It is developed by the Czech company [JetBrains](https://en.wikipedia.org/wiki/JetBrains). It provides code analysis, a graphical debugger, an integrated unit tester, integration with [version control systems](https://en.wikipedia.org/wiki/Revision_control) (VCSes), and supports web development with [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) as well as [Data Science](https://en.wikipedia.org/wiki/Data_science) with [Anaconda](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)).

PyCharm is [cross-platform](https://en.wikipedia.org/wiki/Cross-platform), with [Windows](https://en.wikipedia.org/wiki/Windows), [macOS](https://en.wikipedia.org/wiki/MacOS) and [Linux](https://en.wikipedia.org/wiki/Linux) versions

**Features**

Coding assistance and [analysis](https://en.wikipedia.org/wiki/Code_analysis), with [code completion](https://en.wikipedia.org/wiki/Autocomplete), syntax and error highlighting, [linter integration](https://en.wikipedia.org/wiki/Lint_(software)), and quick fixes.

* Project and code navigation: specialized project views, file structure views and quick jumping between files, classes, methods and usages
* Python [refactoring](https://en.wikipedia.org/wiki/Refactoring): including rename, extract method, introduce variable, introduce constant, pull up, push down and others
* Support for web frameworks: [Django](https://en.wikipedia.org/wiki/Django_(web_framework)) , [web2py](https://en.wikipedia.org/wiki/Web2py) and [Flask](https://en.wikipedia.org/wiki/Flask_(web_framework)).
* Integrated Python debugger.
* [Google App Engine](https://en.wikipedia.org/wiki/Google_App_Engine) Python development
* Version control integration: unified user interface for [Mercurial](https://en.wikipedia.org/wiki/Mercurial), [Git](https://en.wikipedia.org/wiki/Git_(software)), [Subversion](https://en.wikipedia.org/wiki/Apache_Subversion), [Perforce](https://en.wikipedia.org/wiki/Perforce) and [CVS](https://en.wikipedia.org/wiki/Concurrent_Versions_System) with change lists and merge

It competes mainly with a number of other Python-oriented IDEs, including [Eclipse](https://en.wikipedia.org/wiki/Eclipse_(software))'s [PyDev](https://en.wikipedia.org/wiki/PyDev), and the more broadly focused [Komodo IDE](https://en.wikipedia.org/wiki/Komodo_IDE).

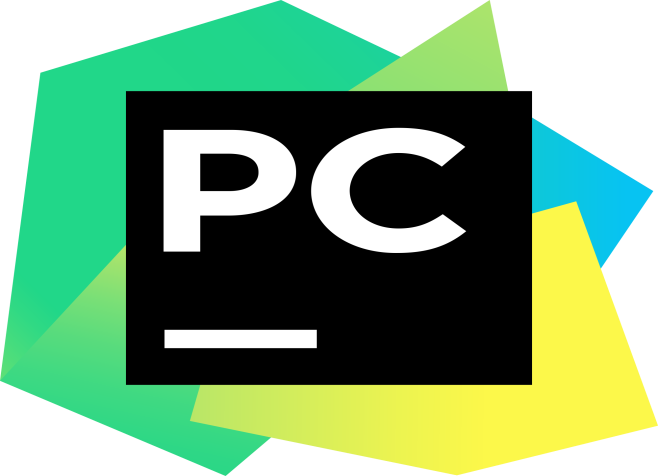


Figure 3

**6. PROPOSED METHODOLOGY (USE CASE DIAGRAM, USE-CASE DIAGRAM**, **DATAFLOW DIAGRAM CHARTS)**

A user case diagram shows you some of the use cases in your system, some of the actors in your system and the relationships between them. A is a high level piece of functionality that the system will provide. An actor is anything or anyone or anything that interacts with the system being built. To build a use case diagram you’ll use a specified set of specialized symbols and connectors.

**The purpose of the use case diagram can be described as follows-**

* Used to gather requirements of the system.
* Used to get an outside view of the system.
* Identify internal and external factors influencing the system.
* Show the interaction between the requirements and the actors.
* The scope of your system
* Goals that your system or application helps those actors achieve.

**6.2 DATA FLOW DIAGRAM**

A data flow diagram is a graphic representation of a system or portion of system. It consists of data flows, processes, sources, destinations, and stores – all described through the use of easily understood Symbols. An entire system can be described from the viewpoint of the data it processes with only four symbol . At the same time, data flow diagrams are powerful enough to show parallel activities.

It is a graphical representation of the & quot; flow & quot; of data through an information system, modeling its process aspects. A DFD is often used as a preliminary step to create an overview of the system, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).A DFD shows what kind of information will be input to and output from the system, where the data will come from and go to, and where the data will bestored.

.

**7. HARDWARE AND SOFTWARE REQUIREMENTS**

**7.1 Technology Stack**

The following technologies have been used for successful implementation of the app ‘ELISA-A Virtual assistant’

* Scikit
* Python
* Octave
* MATLAB
* Pycharm

**7.2 Hardware Requirements:**

* Android Device(4.4 or above)
* Laptop

**7.3 SOFTWARE REQUIREMENTS**

We need to have some software requirements in order to run the app successfully both on the user side as well as the developer side.

**FOR USAGE:**

These requirements are mentioned below:

Android device

User need to have an android device with operating system version 4.4 or above.

This app is only supported by these devices.

**FOR DEVELOPER:**

The developer needs to have the below mentioned requirements for developing and debugging of the app:

**8 . RESULTS**

Few examples to study the project-

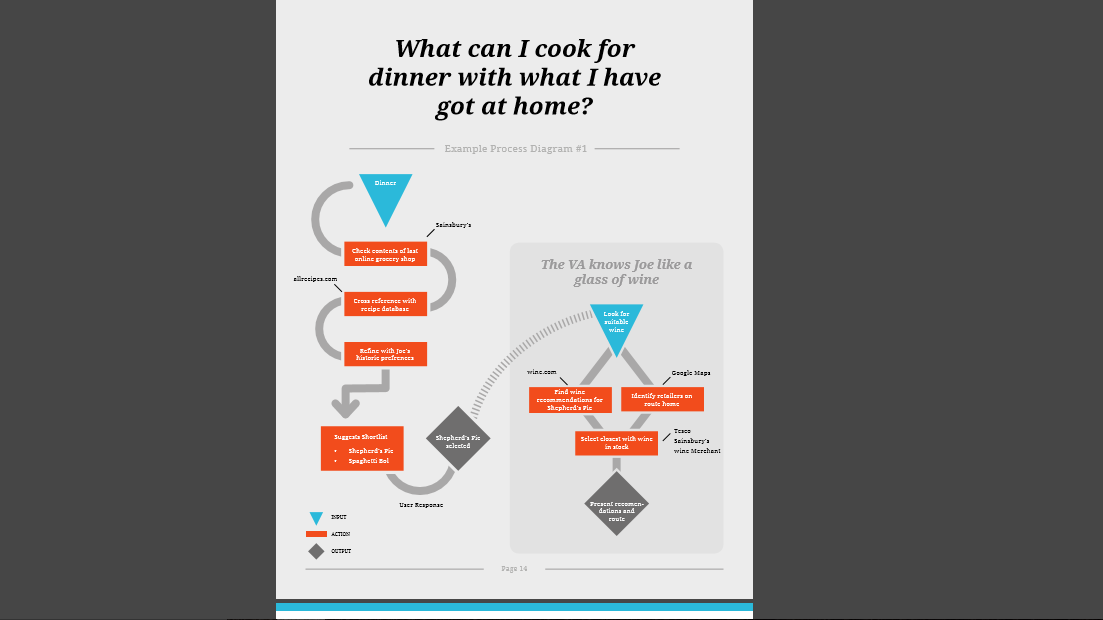
****

Figure 4

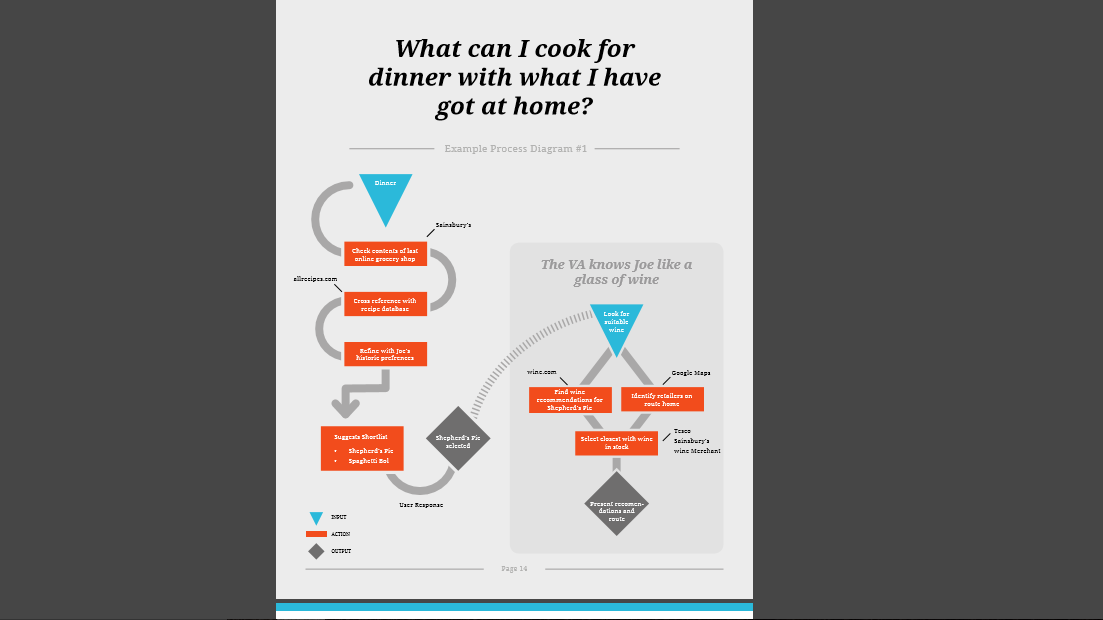
****

Figure 5

Elisa catches the sound of your speech and instantly converts it into a digital form that has stored the relevant information. That signal from your phone will be conveyed wirelessly to a nearby cell tower and through a series of the land line back to your internet service provider where it will communicated with a server, there an algorithm uses machine learning and natural language processing to understand your language and your question.

Concurrently, your speech is classified locally on your phone. The identifier of your device will communicate with the server filled with many languages. The command in the local language can be handled better in comparison to other languages.

The server now measures up your speech with the statistical model and it will consider two things, one is the letters you spoke and the second is the order of your spoken letters. And the highest matched and high probability estimates will go through.

Then the series of vowels and consonants used in your speech are going to make words from it. Those spoken words and order of the words will try to figure out your speech, and estimate it with the statistical model. That can build a confidence level high as the server can easily connect to the speech.

After following the whole process to identify your speech if the server is confident about the result it’ll start responding.

**SNAPSHOTS**

* The ELISA interface
* Working
* Working-Three layers
* Questions to Ask ELISA
* Unique Abilities
* Schematic Flowchart of ELISA
* Speech Processing of ELISA

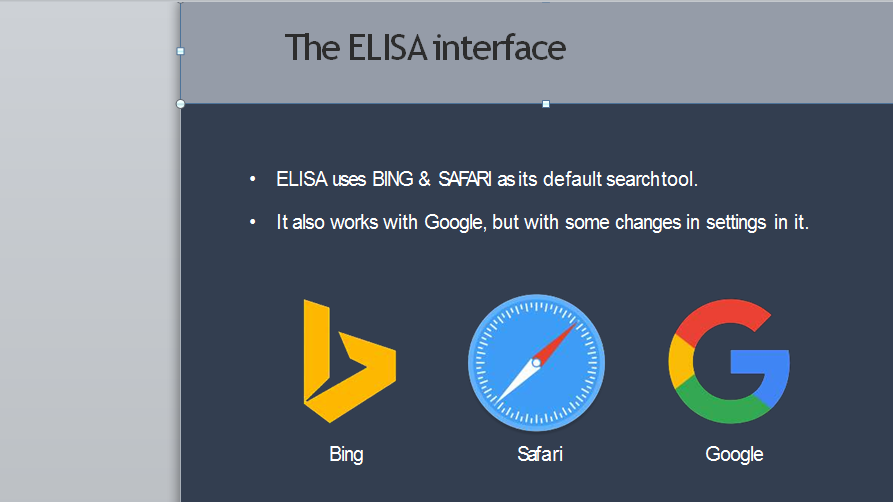
****

Figure 6 (The ELISA Interface)

1. Speech to text:
ï± A Piece of software used that converts audio to text.
It doesnât understand just anything you might s...

Figure 7 (Working)

Name Platform
Google Now Android & IOS
Cortana Windows
Siri IOS
Robin Android
Dragon Go IOS
Evi Android
EasilyDo Android
11
 

Figure 8(Three Layers)

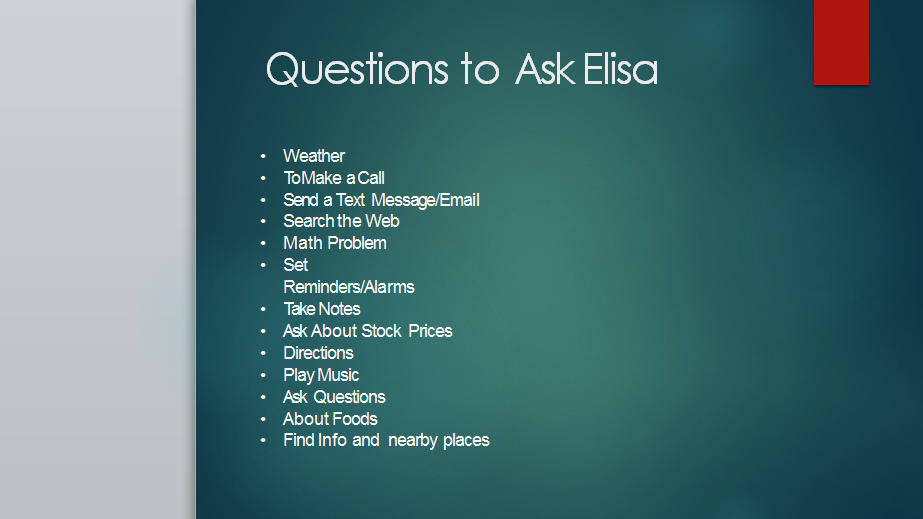
****

Figure 9(Questions to Ask ELISA)

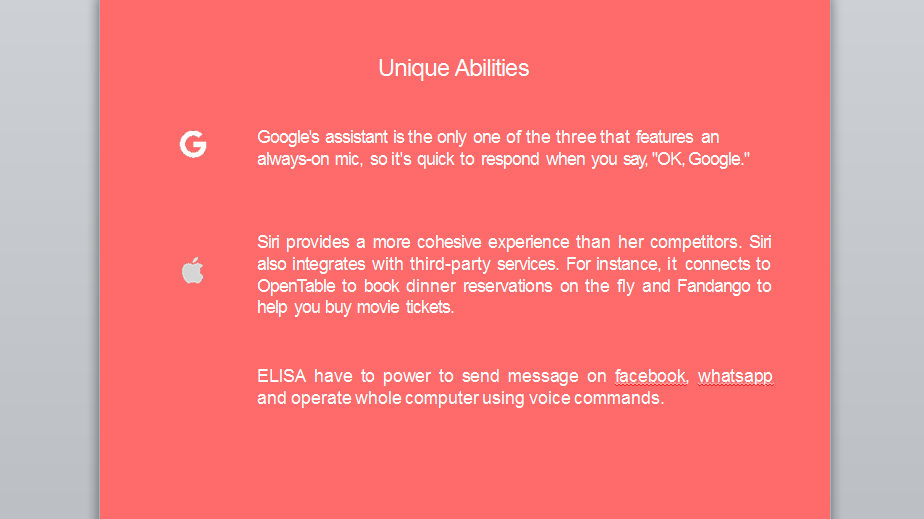
****

Figure 10(Unique Abilities)

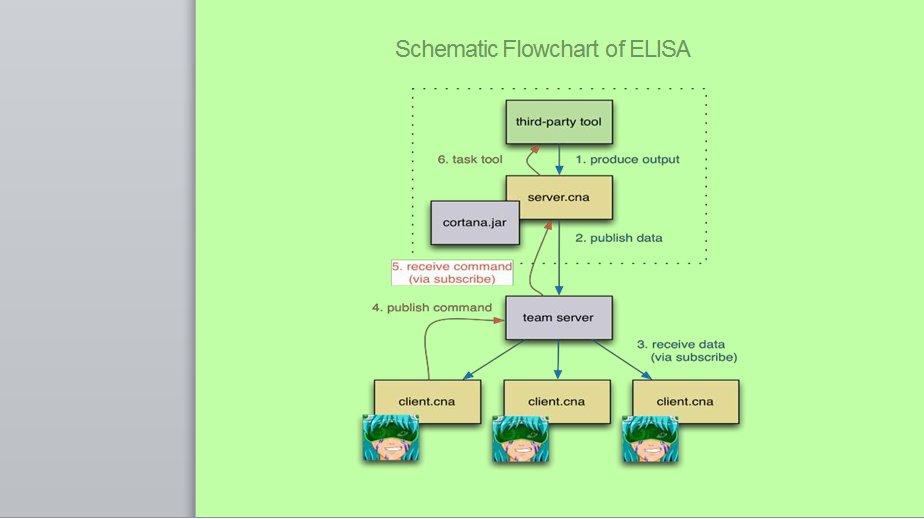
****

Figure 11(Schematic Flowchart of ELISA)

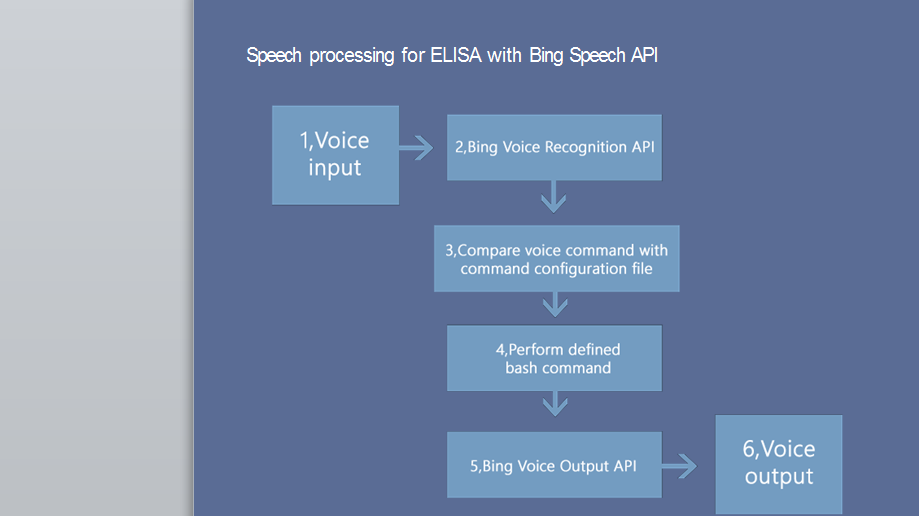
****

Figure 12(Speech Processing)

**9. CONCLUSION**

Thus ELISA is a platform that uses voice queries and a natural-language user interface to answer questions, make recommendations, and perform actions by delegating requests to a set of Internet services. It can access via English.

It can set reminders ,handling device, send messages via various social networking, weather information service, recognize natural voice without the requirement for keyboard input, and answer questions using information from the Google search engine .The speech recognition system makes use of sophisticated machine learning techniques using python.

It's basically voice control that talks back to you, that understands relationships and context, handle basic sequential inference. Ask Elisa questions, or ask Elisa to do things for you, just like you would ask a real assistant, and Elisa will help keep you connected, informed, in the right place, and on time.

* ELISA isn’t just about work, always listening to at you try asking a casual question such as “Where do you come from?”
* The best thing you can do to learn about ELISA is talk to her ,you may be surprised to how she can help you and what she has to say.
* And in the last our vision: We see a world where ELISA will be our personal digital assistant everywhere in our devices:
* If a person login with his/her Microsoft account the personal ELISA with her knowledge about that person, she already knows who is his/her partner, will assist in everyday life trying making it a bit better.