IJARSCT

ISSN (Online) 2581-9429

IJARSCT
Impact Factor: 4.819

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Volume 3, Issue 1, March 2021

Speak2Code: A Multi-Utility Program based on Speech Recognition that Allows you to Code Through Speech Commands

Ronald Maryan Rodriques
M.C.A, Founder Fabulosity, Kolhapur

Abstract: *Voice recognition has gained prominence and extensive use with the rise of Artificial Intelligence and that of the intelligent assistants such as Amazon's Alexa, Apple's Siri and Microsoft's Cortana. Voice recognition systems enable Coders to interact with IDEs, Coding platforms, and tools related to Programming platforms simply by speaking to it. It is also extremely useful for hands free requests, reminders and other simple tasks. In this paper, the researcher has developed a program that simply elevates the use of speech commands to Code efficiently in the Integrated Development Environment and also to handle other related computer activities through the use of a Chat-Bot interface. The researcher has used Microsoft's Speech Engine for Speech Recognition and Synthesis. This Research can prove immensely beneficial for 'Visually Impaired Developers' in the field of Computer Programming by helping them to code and accomplish huge coding possibilities, and challenges before them. This research would prove beneficial with a viewpoint of the concept also called as 'Distant-Coding' or 'Remote-Coding' in which the programmer can code from a remote place without been present in actual at the Company.*

Keywords: Microsoft Speech Engine (MSE), ASR (Automatic Speech Recognition), Speech Commands, VI (Visually-Impaired), Commands, Responses, Integrated Development Environment (IDE)

I. Literature Review

People with visual impairments, use Assistive Technology (AT) like screen readers, screen magnifiers, and braille displays to access computers. They have also been using the same to write computer programs. In recent times, GUI based Integrated Development Environments (IDEs) have become more widely used [1]. These modern IDEs have adopted many innovations to aid program comprehension and development by providing features like syntax highlighting, variable watch windows and ability to execute a code both forward and backward [2]. These feature-rich IDEs enable developers to be more productive and efficient. Though screen readers provide basic accessibility to IDEs, many features that make them so useful to sighted developers remain inaccessible to developers using screen readers. Venkatesh Potluri, Priyan Vaithilingam ET AL in their paper "CodeTalk: Improving Programming Environment Accessibility for Visually Impaired Developers" have grouped the numerous accessibility challenges faced by visually impaired developers in using GUI based programming environments into four categories, namely, discoverability, glanceability, navigability and alertability. They presented CodeTalk, a plugin for Visual Studio that enables Visually Impaired developers to overcome some of these challenges [3]. Jaime Sánchez & Fernando Aguayo in their paper '' Blind Learners Programming through Audio" have introduced APL, Audio Programming Language for blind learners. APL is designed to help novice blind learners to enter to the programming world and to solve problems and develop thinking skills by targeting their needs and mental models [4].

DOI: 10.48175/IJARSCT-827

II. Introduction

Also known as speech recognition, voice recognition is a computer software program or hardware device which has the ability to decode the human voice. Voice recognition is commonly used to control a device, execute commands, or write without having to use a keyboard, mouse, or press any buttons. This all is done today by a computer through automatic speech recognition (ASR) software programs. This ASR program needs to be familiar (i.e. to train the model with his/her voice samples) to the voice commands of the user so that it can more accurately predict, what the user is trying to say or command. For example, you could say "open Chrome" and the computer would open the Chrome Browser. The first known device of Speech Recognition was used in 1952 and it recognized single digits spoken by a user. At this moment, ASR programs can carry out Tremendous and impossible work which is been done in many industries, including Healthcare, Military (Harrier AV-8B & Lockheed Martin F-35), Telecommunications, and personal computing (i.e. hands-free computing).Blind People usually code using a screen reader software. A software programmer at Amazon named Forzano and free Code Camp contributor Florian Beijers use a laptop with a screen reader software, which allows blind or VI users to “read” text through audio cues [5][6]. The researcher’s present work is carried out with the intention of helping these kind of programmers to actually code with their voice commands, rather than interacting with the Hardware, this would be great advantage because:

A. It would lead into less consumption of time used while actually hard coding (typing) the code.

B. It would also result into remote accessing the code platform which would ultimately not need in-person at-place coding.

III. Chat-Bot Architectural Framework

This Chat-bot is developed using C# language and Microsoft Speech Engine for Speech Recognition & Synthesis.This MSE makes available Windows Desktop Speech Technology support for Speech Synthesis Markup Language (SSML) based markup language and the construction of synthetic speech engines. The SSML markup language is the industry standard to provide a rich, XML-based language for assisting the synthetic speech engines. It is endorsed by Microsoft and our competitors. The synthetic speech engine implemented using this Speech Engine can do the following tasks: (A) Receive Input. (B) Queue events, and specify actions (C) Control the control the pitch, speaking rate and volume of the speech output (D) Determine usage and output target of speech synthesis [3].

3.1 Components of MSE Used for the Chat-Bot

Following are the components of MSE used for designing the Chat-bot:

1. Globalization CultureInfo : Provides information about a specific culture (called a locale for unmanaged code development). The information includes the names for the culture. In this case I have used English-Indian.
2. SpeechSynthesizer: SpeechSynthesizer provides access to the functionality of an installed speech synthesis engine.
3. PromptBuilder: PromptBuilder provides methods for adding content,selecting voices and control its attributes.
4. SpeechRecognitionEngine: Instance of this class allows accessing & managing in-process Speech Recognizing Engine.
5. Grammar: Creating object of this class allows to define constraints for speech recognition grammar.

3.2 Program Operation

This Software is designed to recognize the speech, especially of those people who are deprived of eyesight, so that they can use their voice commands for coding. It also has the capabilities for speech recognition and synthesis, means it can convert speech to text and text to speech. The Software User Interface (UI) is divided into two sections a. Input b. Output. When any command is given by the User, it gets populated into Input Section and then the output Tip or Information is generated in the output section, after which the operation is performed. Some responses would be given to the user to suggest that He/She has to give additional command or commands. Some of this functions do not require Internet at all.

DOI: 10.48175/IJARSCT-827

Table A1. shows the user commands to the multi-utility program i.e. Speak2Code which is then processed and the output is derived. The output can be in the form of a task performed or a tip given for further assistance.

SR.NO	CATEGORY	COMMAND	OUTPUTS
1	Miscellaneous	"Open Browser"	"Loading Browser…"
2	Miscellaneous	"Open Calculator"	"Opening Calculator…"
3	Search	"Take me to Google"	"Taking a moment…"
4	Search	"Search for Shivaji University"	"Here are the results for Shivaji University…"
5	Coding	"Open Visual Studio 2012"	"Type by clicking up and down arrow"
6	Coding	"Create New Project"	"Choose the name dialog box"
7	Coding	"Control plus a"	"Please name the project"

Figure 1.1 shows the flow of the Chat-bot in the form of "Use-Case Diagram". The User in this case gives various commands to the Chat-bot. They are processed using the MSE and Synthesizer. After the Command is processed the task is performed or Output is given.

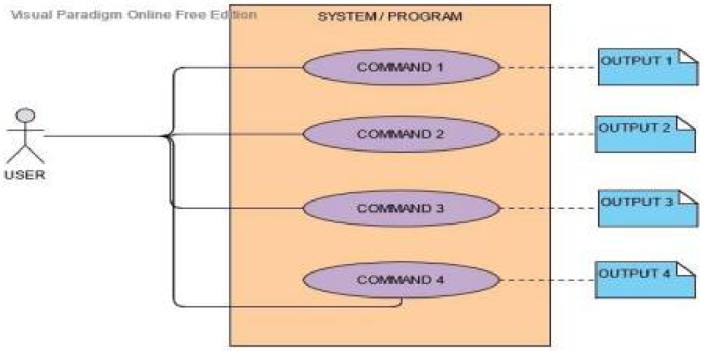
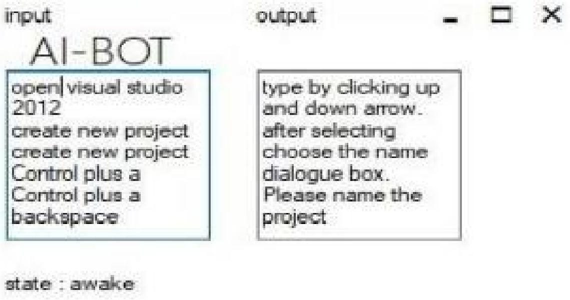


Figure 1.1: Flow of the Program

Figure 1.2 shows the speech commands given to the Chat-Bot. The outputs that are derived are in the form of tasks performed, and other are in the form of some tips given by the Chat-Bot to the user to give or perform further Commands that would help to accomplish the final objective. The Chat-bot's GUI is divided into two sections a. Command or Input b. Response or Output.



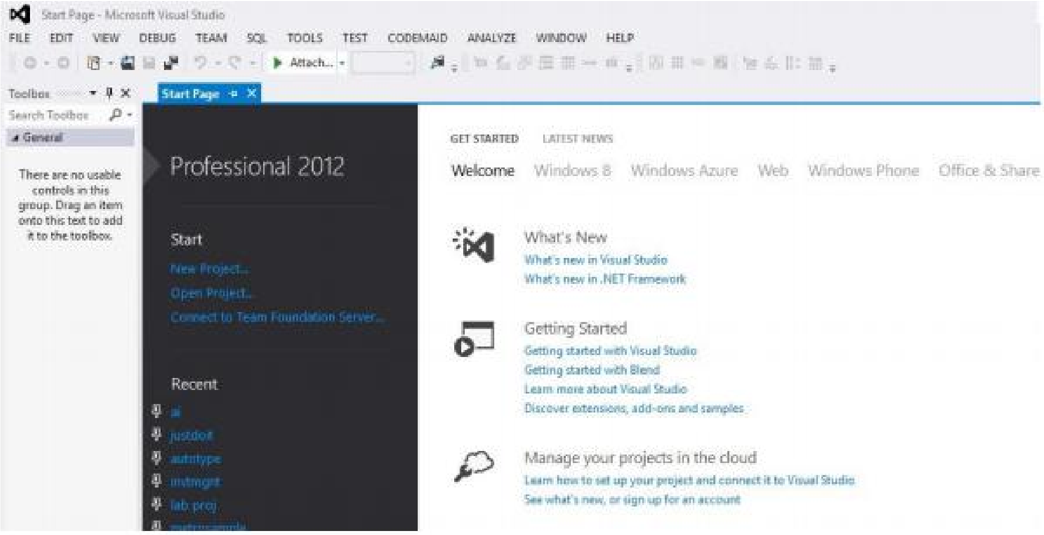
open visual studio
2012
create new project
create new project
Control plus a
Control plus a
backspace

type by clicking up
and down arrow.
after selecting
choose the name
dialogue box.
Please name the
project

state : awake

Figure 1.2: Command given and Responses retrieved from the System

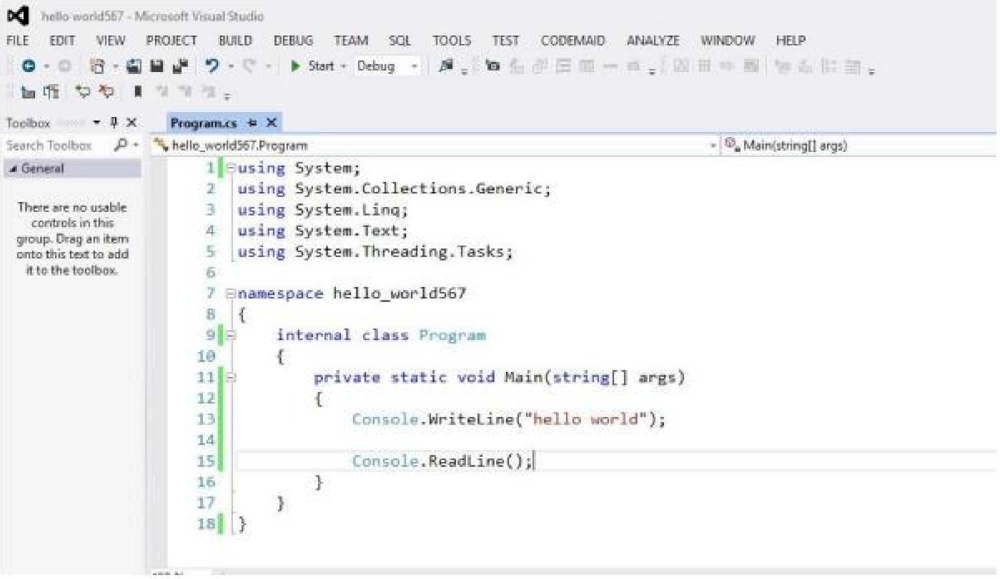
Figure 1.3 shows the task performed after giving the command "Open visual studio 2012".When the User gives the voice command of Opening Visual Studio 2012. It opens the Integrated Development Platform and also gives tips of how to proceed further.


Start Page - Microsoft Visual Studio
FILE EDIT VIEW DEBUG TEAM SQL TOOLS TEST CODEMAID ANALYZE WINDOW HELP
Attach...
Toolbox
Search Toolbox
General
Start Page
Professional 2012
Start
New Project...
Open Project...
Connect to Team Foundation Server...
Recent
ai
justdoit
autotype
intmgnt
lab proj
GET STARTED
LATEST NEWS
Welcome
Windows 8
Windows Azure
Web
Windows Phone
Office & Share
What's New
What's new in Visual Studio
What's new in .NET Framework
Getting Started
Getting started with Visual Studio
Getting started with Blend
Learn more about Visual Studio
Discover extensions, add-ons and samples
Manage your projects in the cloud
Learn how to set up your project and connect it to Visual Studio
See what's new, or sign up for an account

There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.

Figure 1.3: Visual Studio opened after the command 'Open Visual Studio 2012'

Figure 1.4 shows the execution to the command 'backspace' given in the Command Section. All the Code and Text editing functions like clearing the previous word or keyword. Word formatting functions are also added so that every formatting could be done.


hello world567 - Microsoft Visual Studio
FILE EDIT VIEW PROJECT BUILD DEBUG TEAM SQL TOOLS TEST CODEMAID ANALYZE WINDOW HELP
Start
Debug
Toolbox
Search Toolbox
General
Program.cs
hello_world567.Program
Main(string[] args)

There are no usable controls in this group. Drag an item onto this text to add it to the toolbox.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace hello_world567
{
    internal class Program
    {
        private static void Main(string[] args)
        {
            Console.WriteLine("hello world");

            Console.ReadLine();
        }
    }
}
```

Figure 1.4: Execution of the command 'Backspace'

IV. Conclusion

The researcher's system enables Coders and especially those who are VI to code using their speech commands. It also allows remote coding through which one could program from a distant place instead of being present at Company/Office In-person. There is no need of Internet while doing most of the work. Some of the PRO's of this peak2Code System are:

- The User requires no or little guidance on how to use the Application.
- The Application makes you able to handle all text editor functions using voice commands.

DOI: 10.48175/IJARSCT-827

- The Application consumes less time as it operates as a speech to text converter.
- The Application allows you to open different MS Window Softwares, based on voice input.
- The Application provides significant help for people with Disabilities.
- The Application has low Operational Costs.
- All functions work without a Internet Connection expect Google Searches.

Some of the Limitations of this Speak2Code System are:

- Low Accuracy
- Not operational in the Noisy Environment.

V. Future Enhancements

The current Speech Recognition Engine (MSE) does not support a large vocabulary, for this existing problem, the researcher will choose a more sophisticated and extensive Speech Recognition Engine which would have many preinstalled trained samples, more powerful pronunciation identifying Algorithm and accurate speech predicting capacities. The researcher is also planning to move the Entire system to the Cloud Platforms like Amazon's AWS Transcribe or Microsoft Azure's Cognitive Speech Services which are prominent players in Cloud Service Speech providing platform. The next step would be to apply a Machine Learning Algorithm to get more Accurate and Precise responses. Other feature the Researcher would like to roll out, would be the multi-lingual language support which would enable different cultural people from different countries to code in their language. This Research work can be taken into more detail and more work can be done on the research in order to bring advance modifications and additional features.

References

[1]. Stack Overflow. 2017. Stack Overflow Developer Survey 2017. Retrieved September 19, 2017 from https://insights.stackoverflow.com/survey/2017

[2]. Guillaume Pothier, Éric Tanter, and José Piquer. 2007. Scalable omniscient debugging. ACM SIGPLAN Notices 42, 10 (2007), 535–552.

[3]. Venkatesh Potluri, Priyan Vaithilingam, Suresh Iyengar, Y Vidhya, Manohar Swaminathan, Gopal Srinivasa "CodeTalk: Improving Programming Environment Accessibility for VI Developers", CHI '18: Proceedings of the 2018 CHI Conference on Human Factors in Computing SystemsApril 2018 Paper No.: 618 Pages 1–11, https://doi.org/10.1145/3173574.3174192

[4]. Jaime Sánchez & Fernando Aguayo, "Blind Learners Programming through Audio", CHI EA '05: CHI '05 Extended Abstracts on Human Factors in Computing Systems April 2005 Pages 1769–1772 , https://doi.org/10.1145/1056808.1057018

[5]. Neal Karlinsky & Jordan Stead, "Blind since birth, writing code at Amazon since 2013", March 14 2018, Website: https://www.aboutamazon.com/news/workplace/blind-since-birthwriting-code-at-amazon-since-2013

[6]. Quincy Larson, "How blind people code" May 16 2016,Website https://www.freecodecamp.org/news/how-blind-people-code-fdb64e3bf5c/

[7]. Microsoft Speech Engine Documentation by Microsoft. Source: htpps://Docs.Microsoft.Com/Enus /Dotnet/ Api/System.Speech.Recognition.Speech recognition engine.

[8]. Ksenia Shalonova, "Automatic Speech Recognition" 07 DEC 2007 Source:http://www.cs.bris.ac.uk /Teaching/Resources/COMS 12303/lectures /Ksenia_ShalonovaSpeech_Recognition.pdf

[9]. L. Rabiner & B. Juang. 1993. ISBN: 0130151572. "Fundamentals of Speech Recognition".

[10]. D. Jurafsky, J. Martin. 2000. ISBN: 0130950696. Linguistics and Speech Recognition".

[11]. B.H. Juang & Lawrence R. Rabiner, "Automatic Speech Recognition —A Brief History of the Technology Development" 10/08/2004 Source:http://www.ece.ucsb.edu/Faculty/Rabiner/ece259/Reprints/354_LA LI-ASRHistory-final-10-8.pdf

DOI: 10.48175/IJARSCT-827