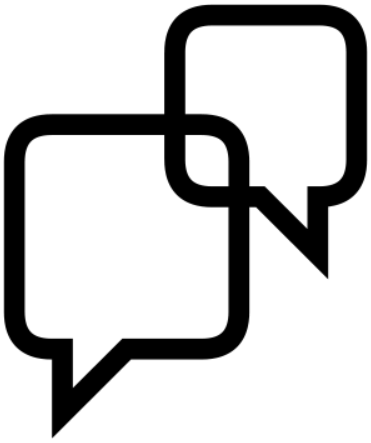
Department of Computer Science & Engineering
The University of Texas at Arlington

Project Charter
CSE 4316: Senior Design I
Summer 2017



Team C
Language Pronounciation Assisting App

Josue C.
Ali S.
Norween J.
Xiwen D.
Kristen R.

Revision History

Revision	Date	Author(s)	Description
0.1	07.03.2017		document creation
0.2	12.13.2017		closeout materials

CONTENTS

List of Figures

1 Vision

Our project vision is to improve current pronounciation-training technology, by providing real-time feedback on word accuracy.

2 Mission

Our mission is to assist people with improving their foreign language pronounciation skills.

3 Success Criteria

A successful scenario for us is one where we've developed a mobile application which gives feedback to the user for their distance to a regular speaker's pronounciation of the same word.

4 Background

Our team has little to no experience in designing mobile applications, and we are required to design a language recognition application that studied the Python libraries of neural networks to instantiate the correct pronunciations of words, the requirement that is stated by the sponsors that the computing logic of the application needs to be computing in the cloud either using Microsoft Azure, Amazon WPS, or Google Cloud, which then will tie into an Android application guided user interface designed using Android Studio and Java and only display the results of the logic.

5 Related Work

Currently there are many Speech and Text Recognition software out in the market, both on iOS and Android platforms, however none of the software achieve what we are trying to achieve. However, we can look at these applications to provide us with a starting point in the initial design of our mobile app. We are trying to appropriate a dictation software that can recognize speech and return the proper pronunciation of the verbiage being used.

6 System Overview

As stated in section 4 (Background) of this document, the requirements are to code logic using python on the cloud where the main computation of the algorithm will concur using the neural network libraries currently present in Python, whereas, after the logic has been computed, the end result, which in this case will be verbiage, with correct pronunciation and context, will be then returned to the GUI designed for Android phones which will be present on the local system (users phone). The main reason being, one cannot be aware of the computing power of an end users phone, if the computing power is less than what is required by the logic then the application itself will crash. A visual representation is below of the idea behind the system design.

7 Roles & Responsibilities

The product owner of our project is Nowreen Jilaney. She will be in charge of making sure that the team knows all of the specifications for the product and will be the lead in interacting with the sponsors. The Scrum Master is Syed Ali. He will be in charge of making sure the team members know what they are working on and keeping the team on track.

8 Facilities & Equipment

As our product is a mobile app, there is no specialized equipment that we need. We will need licenses for the environments we will be using to code the app, such as the environment for Python. We will also need access to a cloud server for computing and storage.

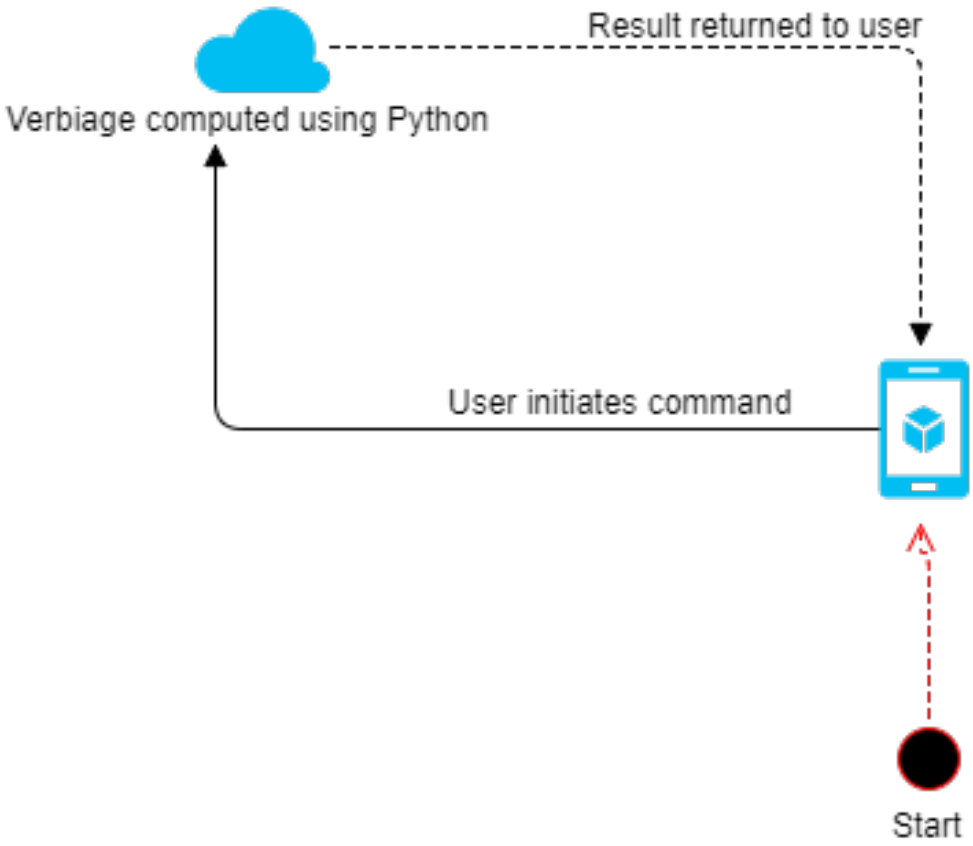


Figure 1: General System Overview

9 Cost Proposal

Details for the preliminary budget, as well as the current & pending support are below.

9.1 Preliminary Budget

The preliminary budget for our project is $800.

9.2 Current & Pending Support

We have not currently spent any of our budget. Our pending spending will include monthly costs for access to the cloud computing service from Amazon Web Service, AWS.

10 Documentation & Reporting

In this section, you will describe all of the various artifacts that you will generate and maintain during the project lifecycle. Describe the purpose of each item below, how the content will be generated, where it will be stored, how often it will be updated, etc.

10.1 Project Charter

Details for the project charter are contained within this document.

10.2 Product Backlog

Task
Investigate Neural Network Python Packages
Create Bare-bones Android Application
Implement server-side code on AWS

10.3 Sprint Work

In this section we detail the work done during each sprint.

10.3.1 Sprint 1

10.3.1.1 Sprint Goal

- Develop an understanding of basic human speech, neural networks, Fourier transform, and Formant frequencies.
- Gather all data and supplies required for Android application development.

10.3.1.2 Sprint Backlog

User Story	Tasks
	Research on basic human speech features
	Research on neural networks
	Research on Fourier transforms and Format frequencies
	Record samples of common US English (enUS) words
1	Learn Javascript and HTML
1	Create a webpage using JavaScript and HTML
1	Design Web Interface
1	Test the program
2	Download proper Android development software
2	Implement recording option for the user
2	Store recorded word
2	Display the word on the screen as a plot
	Research on various python libraries
	Implement Fourier transform, Formant frequency, and spectrogram in Python. Use a speech signal (wo

User Story No.	Description
1	Build an API on mobile platform
1	Build a prototype in Android Studio

10.3.1.3 Task Breakdown

Backlog Item	Est. Time
Research on basic human speech features	12
Research on neural networks	12
Research on Fourier transform and Formant frequencies	4
Record samples of common US English (enUS) words	2
Build an API on a mobile platform to send the input to the server, properly display output	14
Build a prototype in Android studio	12
Research on various Python libraries	2
Implement Fourier transform, Formant frequency, and spectrogram in Python.	10
Use a speech signal (word) as a sample	10
Gather information on cloud development	4

10.3.1.4 Sprint Retrospective

In Sprint 1, we completed our research on basic human speech features, neural networks, and Fourier transform and Formant frequencies. We also researched what we would need our server to do and what environment to work in. We recorded samples of 100 common US English words for the database. We are still working on implementing Fourier transform, Formant frequency, and spectogram. We are also still int eh process of building a prototype of our app.

10.3.2 SPRINT 2

10.3.2.1 Sprint Goal

- Ensure compatibility on all platforms such as Android, Apple, and Windows using Xamarin or other development tools.
- Gather all data and supplies required for Android application development.

10.3.2.2 Sprint Backlog

Backlog Item	Est. Time
Build a prototype in Xamarin or Android Studio	12
Implement Fourier transform, Formant frequency, and spectogram	12
Create SRS documentation	4

10.3.2.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri	
Implement recording option for user	2	1	1	1	0	
Store the recorded words	1	1	0	0	0	
Display the word on the screen as a plot	2	1	1	1	0	
Research on various Python libraries	2	3	2	3	2	
Create SRS documentation	2	2	0	0	0	

10.3.2.4 Sprint Retrospective

In Sprint 2, we completed the SRS documentation and implemented the Fourier transform, Formant frequenct, and spectogram in python. We are still trying to decide if we want to implement the app only in Android Studio or if it should be written in Xamarin so that it can be used by Apple devices as well.

10.3.3 SPRINT 3

10.3.3.1 Sprint Goal

- Design prototype.
- Complete ADS (architectural design specification.

10.3.3.2 Sprint Backlog

Backlog Item	Est. Time
Build a prototype in Xamarin or Android Studio	20
Create ADS documentation	4

10.3.3.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri
Implement recording option for user	2	2	2	2	1
Store the recorded words	1	1	0	0	0
Display the word on the screen as a plot	2	2	2	2	1
Create ADS documentation	2	2	0	0	0

10.3.3.4 Sprint Retrospective

In Sprint 3, we finished designing the prototype and decided to use Android Studio only. This means that our app will only run on Android devices. We also finished our ADS documentation. We are still in the process of building the prototype in Android Studio.

10.3.4 SPRINT 4

10.3.4.1 Sprint Goal

- Expand UML diagram for server-side to include additional details.
- Start implementing sever-side architecture.
- Continue to build the prototype in Android Studio.

10.3.4.2 Sprint Backlog

Backlog Item	Est. Time
Get login information for server from Digital Ocean	5
Build a prototype in Android Studio	10
Finish UML diagram for server-side architecture	7

10.3.4.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri
Finish UML diagram for server-side architecture	2	0	0	0	0
Test SSH and SFTP access	0	1	0	0	0
Set up a dummy web service	1	0	0	0	0
Begin implementing server-side architecture	0	0	2	2	2
Get app to run on phone and connect to server-side dummy web service	2	2	2	2	2

10.3.4.4 Sprint Retrospective

In Sprint 4, we expanded the UML diagram to include all additional details for the server-side architecture. We were also able to test the SSH and SFTP access to make sure they connect to the dummy web service. An empty app was made that only connected to the dummy web service and was able to confirm that it connected. We are still working on implementing the server-side architecture.

10.3.5 SPRINT 5

10.3.5.1 Sprint Goal

- Finish STP documentation.
- Start implementing sever-side architecture.
- Continue to build the prototype in Android Studio and make changes.

10.3.5.2 Sprint Backlog

Backlog Item	Est. Time
Start implementing server-side architecture	6
Continue to build a prototype in Android Studio	10
Finish STP documentation	2

10.3.5.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri	
Start implementing server-side architecture	2	0	2	0	2	
Continue to build prototype in Android Studio	2	2	2	2	2	
Finish STP documentation	0	1	0	1	0	

10.3.5.4 Sprint Retrospective

In Sprint 5, we completed the STP documentation and the server-side architecture. We are still working on building the app in Android Studio.

10.3.6 SPRINT 6

10.3.6.1 Sprint Goal

- Finish implementing server-side architecture.
- Continue to build app in Android Studio.

10.3.6.2 Sprint Backlog

Backlog Item	Est. Time
Finish implementing server-side architecture	4
Continue to build app in Android Studio	12
DDS documentation	2

10.3.6.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri	
Fix bugs in server-side architecture	1	1	1	1	0	
Establish JSON connection to AWS back-end	2	0	2	0	0	
Pass Audio file along with string text of word spoken	0	2	0	0	2	
Return an output in the format of JSON back to the front-end	0	0	0	2	2	
DDS documentation	1	1	0	0	0	

10.3.6.4 Sprint Retrospective

In Sprint 6, we completed the DDS documentation and the server side architecture. We have the app in Android Studio and are still in the process of testing it.

10.3.7 Sprint 7

10.3.7.1 Sprint Goal

- Finish testing the app.
- Test the connection between the server and the application.

10.3.7.2 Sprint Backlog

Backlog Item	Est. Time
Update DDS	2
Application testing	8
Test connection between server and client	6

10.3.7.3 Task Breakdown

Tasks	Mon	Tue	Wed	Thur	Fri	
Update DDS	0	1	0	1	0	
Application testing	2	0	5	1	0	
Test connection between server and client	0	2	0	2	2	

10.3.7.4 Sprint Retrospective

In Sprint 7, we updated the DDS and continue testing the app. There are some minor bugs to work out but we are working on fixing them. The connection between the server and the client is working.

10.4 Closeout Materials

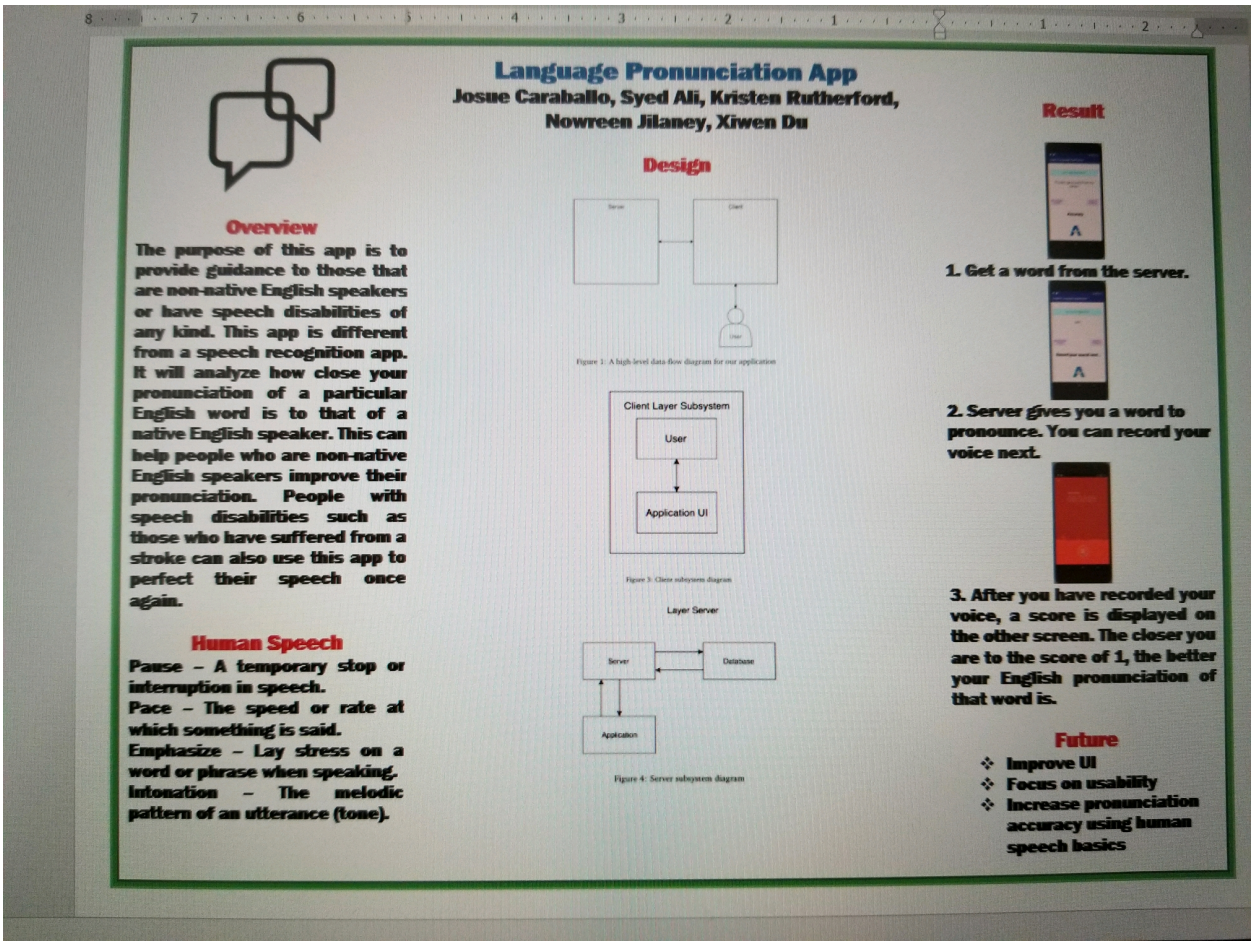
10.4.1 Project Poster

10.4.2 Web Page

Our website is located here.

10.4.3 Demo Video

The demo video for our app is located on our website under the Preview tab.



10.4.4 Installation Scripts

To install the app go to the Google Plus store to download it onto your Andriod phone.