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

Architectural Design Specification
CSE 4316: Senior Design I
Summer 2017

Team C
Language Pronounciation Assisting App

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

Revision History

Revision	Date	Author(s)	Description
0.1	8.8.2017	GH	document creation

CONTENTS

List of Figures

List of Tables

Introduction

This section describes the purpose, use and intended user audience for the Language Pronunciation App. The Language Pronunciation App is an application that helps users improve their pronunciation of phonetically difficult words.

Users will be able to visualize their distance to the “perfect“ phonetic pronunciation of a word.

System Overview

In order to minimize the burden of processing on the individual clients that utilize our system the application takes a lightweight-client approach to the traditional client-server application.

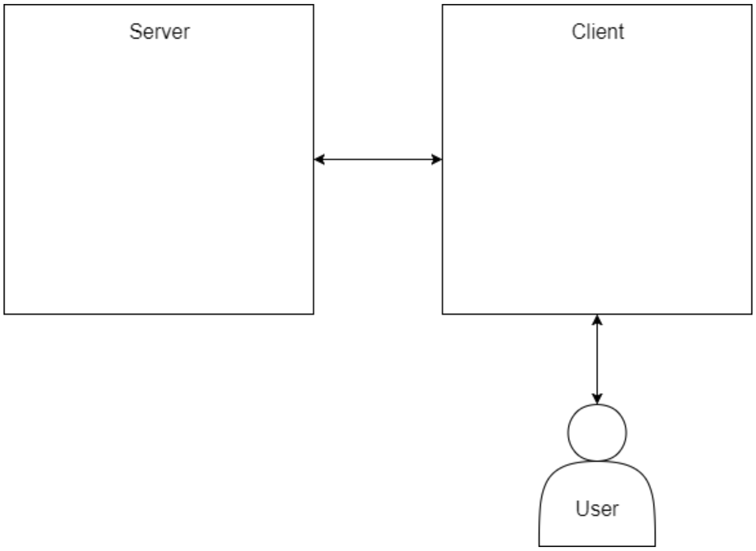

Figure 1: A high-level data-flow diagram for our application

Client Layer

The client will function as the HMI between our system and the user. It consists of a UI, and methods by which it may communicate with the server.

Server Layer

The server will function as the work-horse for the application. It's purpose is to host the visualization function which will map an input of word, audio to a distance metric in the form of x, y co-ordinates.

SUBSYSTEM DEFINITIONS & DATA FLOW

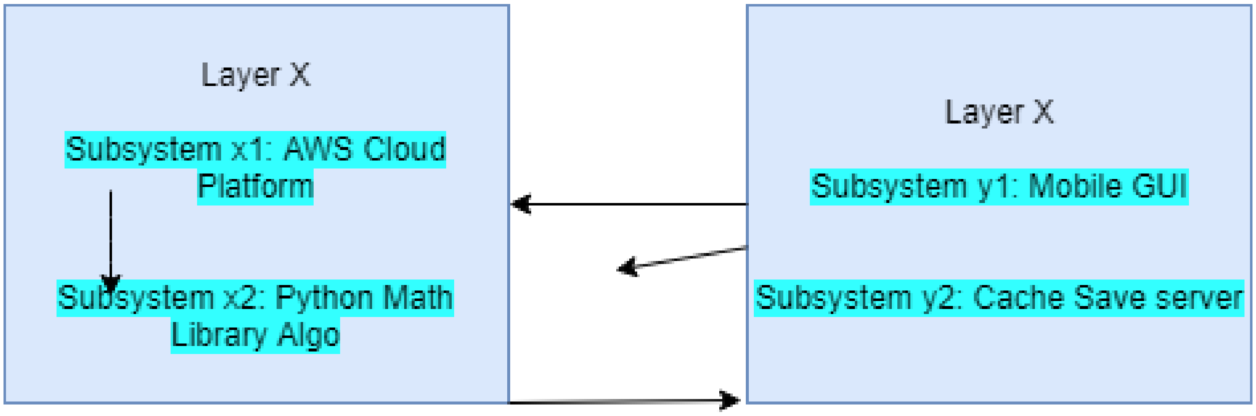
Layer X
Subsystem x1: AWS Cloud Platform
Subsystem x2: Python Math Library Algo
Layer X
Subsystem y1: Mobile GUI
Subsystem y2: Cache Save server

Figure 2: The data flow diagram for our app

Client Layer Subsystems

Client Subsystem

The client layer subsystem will display the UI of the application. It has a XY graph and a dot calculator. It has a button for record the voice of the users

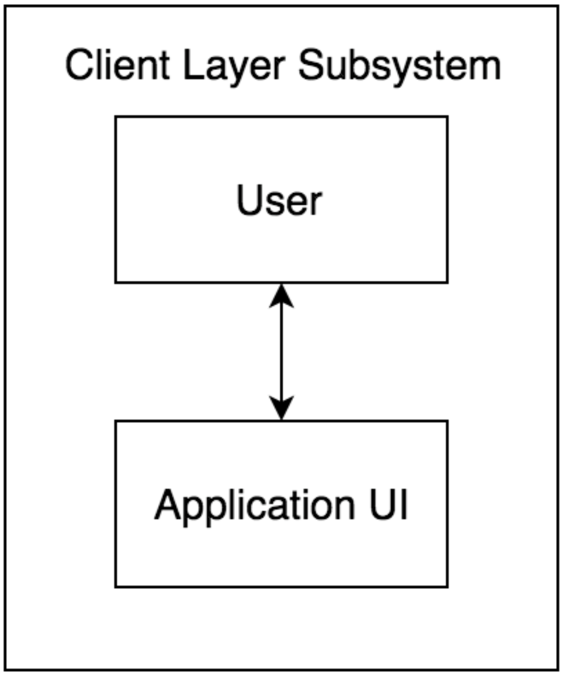

Figure 3: Client subsystem diagram

Assumptions

The users press the record button, it will be recorded the sound and display the accurate dot.

Responsibilities

The Client Layer Subsystem will send the user's voice to the application, and interaction with user. And it will display a dot on the application UI. The Client Layer should display all information to user. And receive user's request.

Subsystem Interfaces

Each of the inputs and outputs for the subsystem are defined here. Create a table with an entry for each labelled interface that connects to this subsystem. For each entry, describe any incoming and outgoing data elements will pass through this interface.

Table 2: Client Subsystem interfaces

ID	Description	Inputs	Outputs
#1	XY Graph	N/A	Red dot
#2	Voice Record Button	User’s sound	N/A

Server Layer Subsystems

Server Subsystem

The system records the users voice sound and send it to sever. Sever compare the sound with the database and calculate the difference and send back to the application.

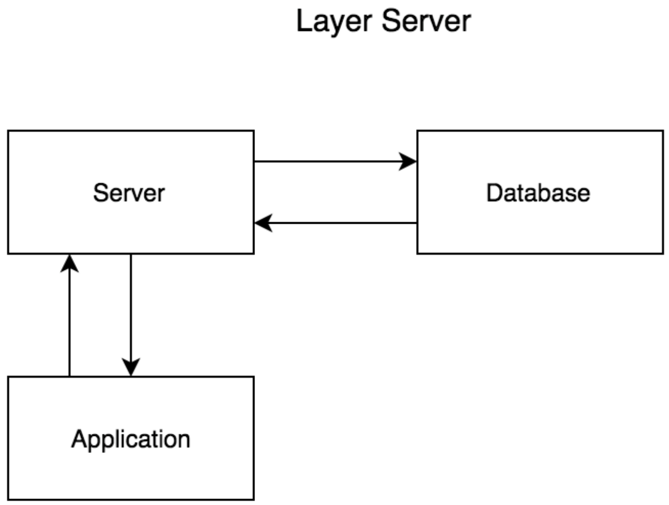

Figure 4: Server subsystem diagram

Assumptions

The Server receive the user's voice data and compare data with the standard voice data from the database.

Responsibilities

The application sends the voice data to the server. The sever get data from the database and calculate the difference, then send the information back. The application records the user's sound, and send to the server, and when sever received the user's voice data, then compare data with the standard voice data from the database. The server send back the % difference to the application.

Subsystem Interfaces

Table 3: Server Subsystem interfaces

ID	Description	Inputs	Outputs
#1	Display the graph dot by calculate the difference of the voice words	Voice Words	Graph Dot

REFERENCES