Jake Mathews

Lab 8

Problem Statement

The problem is to be able to record a user's voice and translate that into text. The user must also be able to type words and have the device read aloud the text back to them.

Analysis

I will create a mobile application with two button and an editable text field. The first button will trigger a dialog box to pop up that can record a person's voice. When they are done talking, the translated Speech-To-Text will appear in a text field below the button. The second button will be disabled by default with some helpful text below it. Below that text will be an editable text field in which the user can type what they want to be read aloud to them. If there is at least one character in the editable text field, the second button will become enabled. When the user presses that button, the system will read aloud whatever text is in the editable text field.

Some difficulties were in testing because the Android emulator cannot record audio or play back audio.

Jake Mathews

Screen Shots

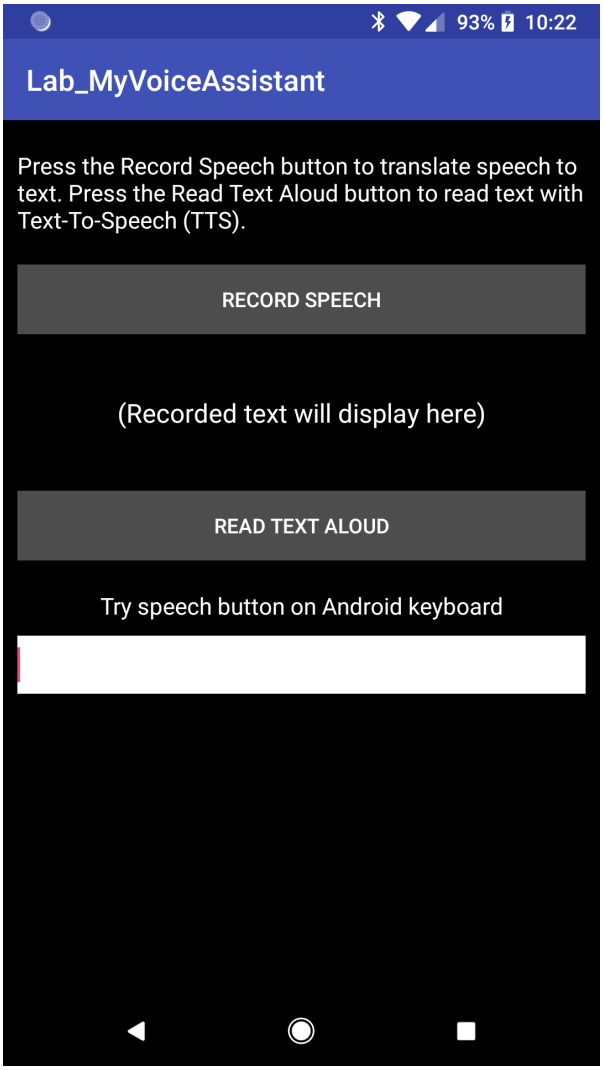

Figure 1: Application on startup

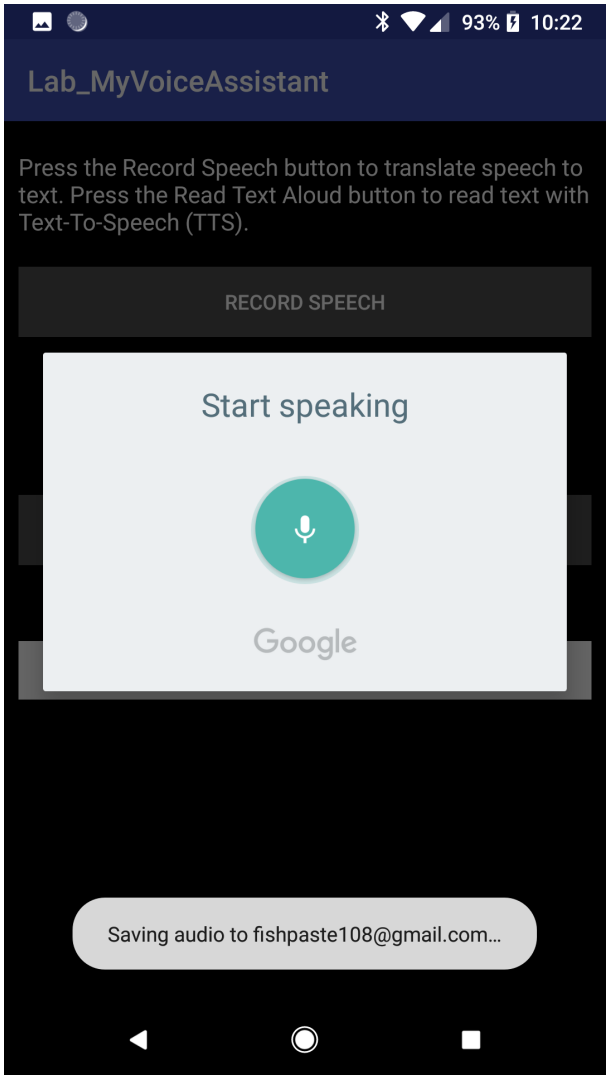

Figure 2: Dialog box that appears after pressing the first button (RECORD SPEECH)

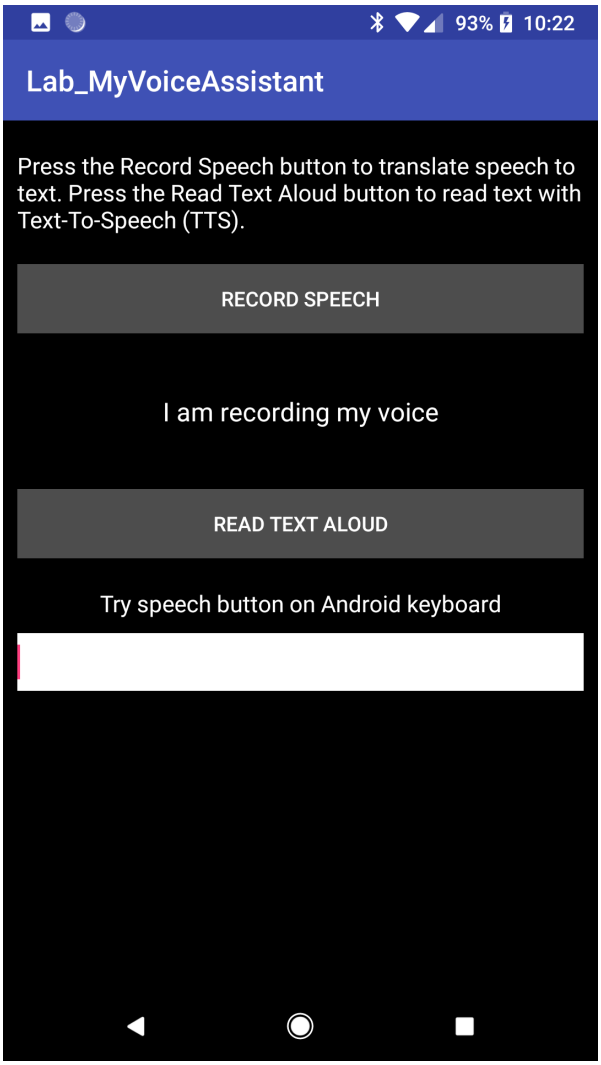

Figure 3: The speech that was said during the dialog now appears as text below the RECORD SPEECH button

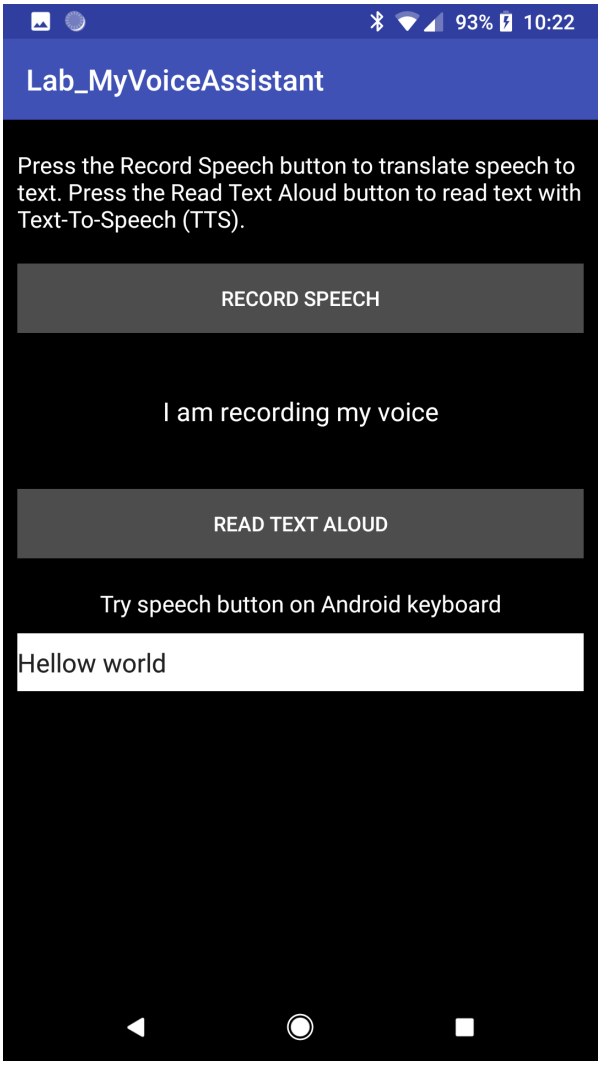

Figure 4: The user can enter text into the text field. When they press the READ TEXT ALOUD button, the text that will be read aloud is the contents of that text field.

Source Code

Manifests/AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="edu.wit.mobileapp.lab8_voice">

    <uses-feature android:name="android.hardware.microphone" />
    <uses-permission android:name="android.permission.INTERNET"
/>
    <uses-permission
android:name="android.permission.RECORD_AUDIO" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

java/ edu/wit/mobileapp/lab8_voice/MainActivity.java

```
package edu.wit.mobileapp.lab8_voice;

import android.content.ActivityNotFoundException;
import android.content.Intent;
import android.os.Bundle;
import android.speech.RecognizerIntent;
import android.speech.tts.TextToSpeech;
import android.support.v7.app.AppCompatActivity;
import android.text.Editable;
import android.text.TextWatcher;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {
    final int REQUEST_CODE = 100;
    final String SPEECH_TAG = "SPEECH";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        final Button speechToTextBtn = (Button)
findViewById(R.id.ButtonRecord);
        final Button textToSpeechBtn = (Button)
findViewById(R.id.ButtonRead);
        final EditText editTextField = (EditText)
findViewById(R.id.editTextField);

        // Speech To Text
        speechToTextBtn.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(final View view) {
                final Intent intent = new
Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
```

```
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
                intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
Locale.getDefault());
                intent.putExtra(RecognizerIntent.EXTRA_PROMPT,
"Start speaking");
                try {
                    startActivityForResult(intent,
REQUEST_CODE);
                    Log.i(SPEECH_TAG, "Listening");
                } catch (ActivityNotFoundException exception) {
                    Log.e(SPEECH_TAG, "Could not start
activity", exception);
                }
            }
        });

        // Text to Speech
        editTextField.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence
charSequence, int i, int i1, int i2) {
                // Do nothing. We don't care what the text was
before.
            }

            @Override
            public void onTextChanged(CharSequence charSequence,
int i, int i1, int i2) {
                if (charSequence.toString().isEmpty()) {
                    textToSpeechBtn.setEnabled(false);
                } else {
                    textToSpeechBtn.setEnabled(true);
                }
            }

            @Override
            public void afterTextChanged(Editable editable) {
                // Do nothing. The state of the button is set
anytime the text is changed. Not just when the user is done.
            }
        });

        final TextToSpeech textToSpeech = new
TextToSpeech(getApplicationContext(), new
TextToSpeech.OnInitListener() {
```

```
            @Override
            public void onInit(final int status) {
                if(status == TextToSpeech.ERROR) {
                    Log.e(SPEECH_TAG, "Failed to initialize
Text-To-Speech");
                }
            }
        });
        textToSpeech.setLanguage(Locale.getDefault());

        textToSpeechBtn.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(final View view) {
                Log.i(SPEECH_TAG, "Reading aloud");
                final String text =
editTextField.getText().toString();
                textToSpeech.speak(text,
TextToSpeech.QUEUE_FLUSH, null,
String.valueOf(this.hashCode()));
            }
        });
    }

    @Override
    // Called when the speech to text activity finishes
    protected void onActivityResult(int requestCode, int
resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        final TextView textSaidView = (TextView)
findViewById(R.id.TextSaid);
        switch (requestCode) {
            case REQUEST_CODE: {
                if (resultCode == RESULT_OK && null != data) {
                    final ArrayList<String> result =
data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
                    textSaidView.setText(result.get(0));
                }
                break;
            }
        }
    }
}
```

res/layout/activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android
        ="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:paddingLeft="10dp"
    android:paddingRight="10dp"
    android:gravity="top"
    android:orientation="vertical"
    android:background="#000000"
    >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:textColor="#ffffff"
        android:textSize="16sp"
        android:text="@string/help" />

    <Button
        android:id="@+id/ButtonRecord"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="@string/button_record"
        android:textColor="#ffffff"
        android:background="#4d4d4d" />

    <TextView
        android:id="@+id/TextSaid"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:lines="5"
        android:text="@string/recorded_text"
        android:textSize="18sp"
        android:textColor="#ffffff"/>

    <Button
        android:id="@+id/ButtonRead"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
```

```
        android:enabled="false"
        android:text="@string/button_read"
        android:textColor="#ffffff"
        android:background="#4d4d4d" />

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:gravity="center_horizontal"
        android:textSize="16sp"
        android:textColor="#ffffff"
        android:text="@string/try_help"/>

    <EditText
        android:id="@+id/editTextField"
        android:layout_width="fill_parent"
        android:layout_height="40dp"
        android:layout_marginTop="10dp"
        android:inputType="textCapSentences"
        android:background="#ffffff"/>
</LinearLayout>
```

res/values/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android
        ="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:paddingLeft="10dp"
    android:paddingRight="10dp"
    android:gravity="top"
    android:orientation="vertical"
    android:background="#000000"
    >

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:textColor="#ffffff"
        android:textSize="16sp"
        android:text="@string/help" />

    <Button
        android:id="@+id/ButtonRecord"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:text="@string/button_record"
        android:textColor="#ffffff"
        android:background="#4d4d4d" />

    <TextView
        android:id="@+id/TextSaid"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:lines="5"
        android:text="@string/recorded_text"
        android:textSize="18sp"
        android:textColor="#ffffff"/>

    <Button
        android:id="@+id/ButtonRead"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
```

```
        android:enabled="false"
        android:text="@string/button_read"
        android:textColor="#ffffff"
        android:background="#4d4d4d" />

    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:gravity="center_horizontal"
        android:textSize="16sp"
        android:textColor="#ffffff"
        android:text="@string/try_help"/>

    <EditText
        android:id="@+id/editTextField"
        android:layout_width="fill_parent"
        android:layout_height="40dp"
        android:layout_marginTop="10dp"
        android:inputType="textCapSentences"
        android:background="#ffffff"/>
</LinearLayout>
```