Anusha Nelluri – anhx2@umsystem.edu

GitHub Link - https://github.com/NelluriAnusha/Demo_Remote/tree/main/Mobilepart/ICP11

Achyuth Kumar Valeti – avgh3@umsystem.edu

GitHub Link - https://github.com/AchyuthValeti/Demo_Remote/tree/main/Mobilepart/ICP11

ICP11 (Text-to-Speech)

A. Introduction:

Android Studio is an open-source framework for developing apps for Android OS devices including smartphones, tablets, and televisions. In this task, we are using Text to speech, and it converts text on the screen into speech. Text-to-speech is a popular accessibility function that assists those who have difficulty reading on-screen text, but it's also useful for those who wish to be read to. This function has proven to be a popular and beneficial feature among users.

In this ICP, we used **Android Studio, Java & XML** languages for mobile application development.

B. Task Description:

In this task, we were asked to use Text-to-Speech functionality to convert the text that is entered on the screen into speech using android studio.

C. Implementation Process:

- In Android studio application, we have created a new project and selected "Empty activity".
- Empty activity provides .java and .xml files (MainActivity.java, activity_main.xml) and it is used for converting typed text into speech.

1. **'activity_main.xml' File (Creating a Button, TextView & EditText fields):**

    a. In 'activity_main.xml' file, we have added TextView, EditText and Button fields to get the input from the user and convert it into speech.
    b. Using EditText field, we can add text message in the application which means it helps us to input the text from user.

```
<EditText
    android:id="@+id/editTextTTS"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:hint="Enter your text"
    android:layout_below="@+id/textViewTTS"
    android:layout_margin="35dp"
    android:layout_centerHorizontal="true" />
```

c. Button field helped to convert the typed text into a speech. When “Speak” button is clicked, it will convert text into speech.

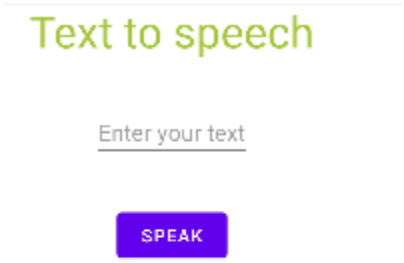
```
<Button
    android:id="@+id/btnTTS"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/editTextTTS"
    android:text="Speak"
    android:layout_centerHorizontal="true" />
```

d. Using TextView field, we have printed the “Text to speech” message on the screen. So, it is used to display on the screen.

```
<TextView
    android:id="@+id/textViewTTS"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Text to speech"
    android:textSize="35dp"
    android:textColor="#a4c639"
    android:layout_centerHorizontal="true" />
```

e. We have added all the properties of height, width, textSize & textColor to get the mobile page in a proper alignment.

f. **Output**:



2. **‘MainActivity.java’ File (Adding Text-to-Speech functionality to convert text into speech):**

a. Firstly, we have initialized EditText & Button field values with the values from id’s in the ‘activity_main.xml’ file.
b. Then, we have created an object named it as ‘tts’ and added all Text-to-speech features to that object to get the complete functionality.
c. Once the object creation is done, we have added validations for both success and failure cases.

d. If no error is found, then it will run successfully and selected the language of speech as “Locale.US”.
e. We have added validations for language selection as well.
f. When the mentioned language is not correct or not working properly, then error message will be displayed. Otherwise, text will be converted successfully.

```
public void onClick(View view) {
    tts = new TextToSpeech( context: MainActivity.this, new TextToSpeech.OnInitListener() {
        @Override
        public void onInit(int status) {
            if(status == TextToSpeech.SUCCESS) {
                int result = tts.setLanguage(Locale.US);
                if (result == TextToSpeech.LANG_NOT_SUPPORTED || result == TextToSpeech.LANG_MISSING_
                    Log.e( tag: "message",  msg: "Language is not supported");
                } else {
                    speak();
                }
            } else{
                Log.e( tag: "message",  msg: "TTS is not supported");
            }
```

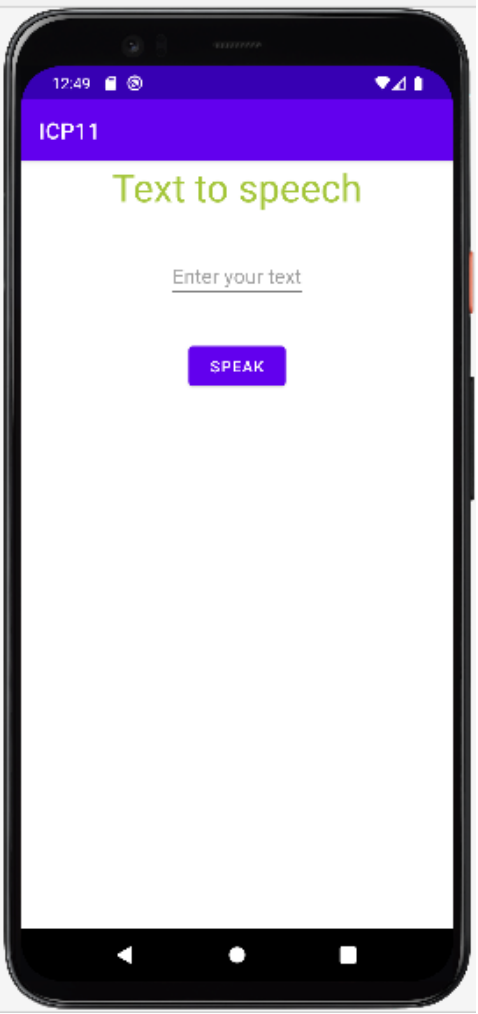
g. We used “stop()” function to interrupt the current sentence (whether it's being played or saved to a file) and discard the rest of the queue's sentences.

```
protected void onPause(){
    super.onPause();
    tts.stop();
}
```

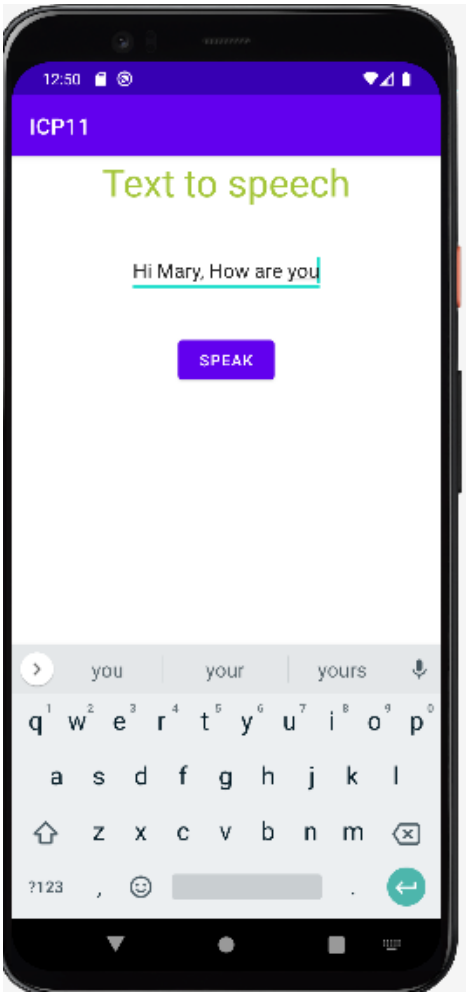
h. Here, we also used shutdown() function and it is used to release the TextToSpeech native resources.

```
protected void onDestroy(){
    super.onDestroy();
    tts.stop();
    tts.shutdown();
}
```

D. Output Page in Android Mobile:



Need to enter the text then click on "SPEAK" Button to get it in the voice.



When we enter the text and click on "SPEAK" button, the text is converted to Voice(speech). We can hear the voice.

E. Contribution:

We have contributed equally.

F. Conclusion:

In this ICP, we have learned Text-to-Speech functionality to convert the text entered on the screen into Speech(voice) and its features and developed a mobile android application using the same.

G. Challenges:

We have not faced any major challenges while doing the assignment.