EX:NO:13 TEXT TO SPEECH REG.NO:210701515

DATE:3/5/2024

AIM:-

Develop an android application to perform Text to Speech.

PROCEDURE:-

Step 1: Create a new Android Project.

Step 2: Add text-to-speech dependency.

Step 3: Design the user interface.

Step 4: Initialize text-to-speech engine.

Step 5: Implement text-to-speech functionality.

Step 6: Test the application.

Step 7: Handle lifecycle events.

Step 8: Optimize and refine.

PRGRAM CODE:-

AndroidManifest.xml:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.texttospeech">
    <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>
```

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:hint="Enter text to speak"
        android:layout_margin="16dp" />
    <Button
        android:id="@+id/buttonSpeak"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Speak"
        android:layout_below="@id/editText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp" />
</RelativeLayout>
```

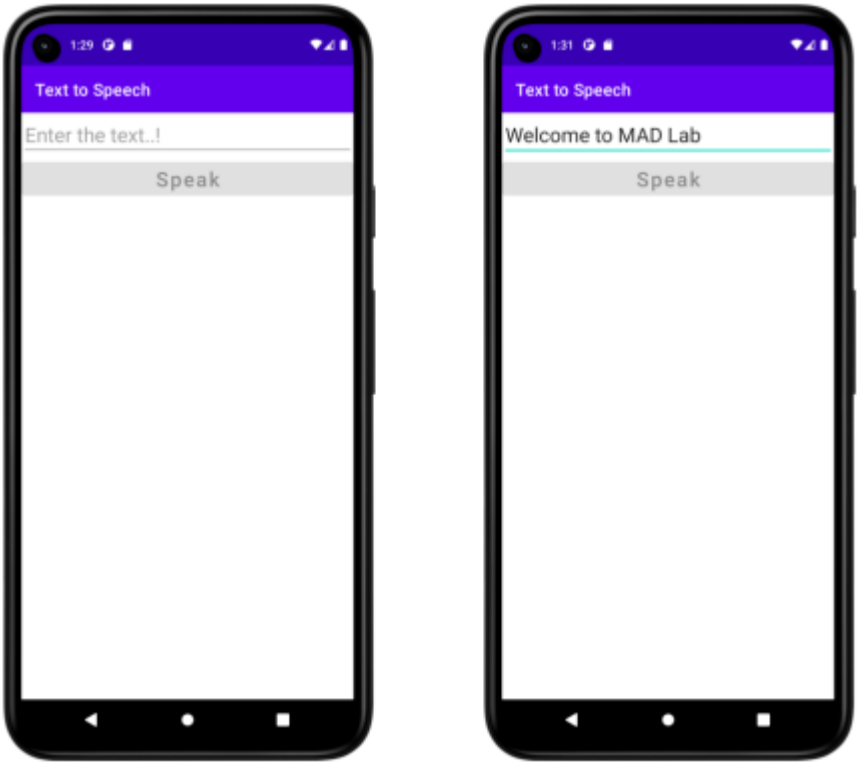
MainActivity.kt:

```
package com.example.texttospeech
import android.os.Bundle
import android.speech.tts.TextToSpeech
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import java.util.*
class MainActivity : AppCompatActivity(), TextToSpeech.OnInitListener {
    private lateinit var textToSpeech: TextToSpeech
    private lateinit var editText: EditText
```

```
    private lateinit var buttonSpeak: Button
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        editText = findViewById(R.id.editText)
        buttonSpeak = findViewById(R.id.buttonSpeak)
        textToSpeech = TextToSpeech(this, this)
        buttonSpeak.setOnClickListener {
            val text = editText.text.toString()
            if (text.isNotEmpty()) {
                speak(text)
            } else {
                Toast.makeText(this, "Please enter some text",
Toast.LENGTH_SHORT).show()
            }}}
    override fun onInit(status: Int) {
        if (status == TextToSpeech.SUCCESS) {
            val result = textToSpeech.setLanguage(Locale.US)
            if (result == TextToSpeech.LANG_MISSING_DATA || result ==
TextToSpeech.LANG_NOT_SUPPORTED) {
                Toast.makeText(this, "Language not supported",
Toast.LENGTH_SHORT).show()
            }} else {
```

```
            Toast.makeText(this, "Initialization failed",
Toast.LENGTH_SHORT).show()
        }}
    private fun speak(text: String) {
        textToSpeech.speak(text, TextToSpeech.QUEUE_FLUSH, null, "")}
    override fun onDestroy() {
        if (textToSpeech.isSpeaking) {
            textToSpeech.stop()
        }
        textToSpeech.shutdown()
        super.onDestroy()
    }
}
```

OUTPUT:-



RESULT:-

Thus to develop an android application to perform Text to Speech is implemented and executed successfully.