EX 14

ANDROID MANIFEST

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

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

    <application
            android:allowBackup="true"
            android:dataExtractionRules="@xml/data_extraction_rules"
            android:fullBackupContent="@xml/backup_rules"
            android:icon="@mipmap/ic_launcher"
            android:label="@string/app_name"
            android:roundIcon="@mipmap/ic_launcher_round"
            android:supportsRtl="true"
            android:theme="@style/Theme.Ex14"
            tools:targetApi="31">
        <activity
                android:name=".MainActivity"
                android:exported="true">
            <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">

    <ImageButton
            android:id="@+id/micButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:src="@android:drawable/ic_btn_speak_now"/>

    <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/micButton"
            android:layout_centerHorizontal="true"
            android:layout_marginTop="20dp"
            android:text="Output appears here..."
            android:textSize="20sp" />

</RelativeLayout>
```

MAIN ACTIVITY

```kotlin
package com.example.ex14

import android.Manifest
import android.content.Intent
import android.content.pm.PackageManager
import android.speech.RecognitionListener
import android.speech.RecognizerIntent
import android.speech.SpeechRecognizer
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.ImageButton
import android.widget.TextView
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
import java.util.*

class MainActivity : AppCompatActivity() {

    private lateinit var micButton: ImageButton
    private lateinit var textView: TextView
    private lateinit var speechRecognizer: SpeechRecognizer

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        micButton = findViewById(R.id.micButton)
        textView = findViewById(R.id.textView)

        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.RECORD_AUDIO)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions(
                this,
                arrayOf(Manifest.permission.RECORD_AUDIO),
                1
            )
        }

        micButton.setOnClickListener {
            startListening()
        }

        speechRecognizer = SpeechRecognizer.createSpeechRecognizer(this)
        speechRecognizer.setRecognitionListener(object : RecognitionListener {
            override fun onReadyForSpeech(params: Bundle?) {}
            override fun onBeginningOfSpeech() {}
            override fun onRmsChanged(rmsdB: Float) {}
            override fun onBufferReceived(buffer: ByteArray?) {}
            override fun onEndOfSpeech() {}
            override fun onError(error: Int) {
                textView.text = "Error: $error"
            }
            override fun onResults(results: Bundle?) {
                val result =
results?.getStringArrayList(SpeechRecognizer.RESULTS_RECOGNITION)
                textView.text = result?.get(0) ?: ""
            }
            override fun onPartialResults(partialResults: Bundle?) {}
            override fun onEvent(eventType: Int, params: Bundle?) {}
        })
    }
```

```kotlin
    private fun startListening() {
        val intent = Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH)
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
            RecognizerIntent.LANGUAGE_MODEL_FREE_FORM)
        intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE, Locale.getDefault())
        speechRecognizer.startListening(intent)
    }

    override fun onDestroy() {
        super.onDestroy()
        speechRecognizer.destroy()
    }
}
```