

Androidmanifest.xml

```
<manifest
xmlns:android="http://schemas.android.com/apk/res/andr
oid"
package="com.example.exe05">
<uses-permission
android:name="android.permission.INTERNET" />
<uses-permission
android:name="android.permission.RECORD_AUDIO"
/> <uses-permission
android:name="android.permission.WRITE_EXTERNAL
_STORAGE" /> <uses-permission
android:name="android.permission.READ_EXTERNAL
_STORAGE" />
<application android:allowBackup="true"
android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
android:roundIcon="@mipmap/ic_launcher_round"
android:supportRtl="true"
android:theme="@style/AppTheme">
<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>
```

MainActivity.java

```
package com.example.exe05;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
private TextToSpeechHelper textToSpeechHelper;
private SpeechToTextHelper speechToTextHelper;
private EditText editText;
@Override protected void onCreate(Bundle
savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
editText = findViewById(R.id.editText);
Button buttonSpeak = findViewById(R.id.buttonSpeak);
Button buttonListen = findViewById(R.id.buttonListen);
Button buttonClear = findViewById(R.id.buttonClear);
textToSpeechHelper = new TextToSpeechHelper();
textToSpeechHelper.initTextToSpeech(this);
speechToTextHelper = new SpeechToTextHelper();
speechToTextHelper.initSpeechToText(this);
speechToTextHelper.setOnSpeechRecognitionListener(r
ecognizedText -> {
editText.setText(recognizedText);
textToSpeechHelper.speakText("You said: " +
recognizedText); });
buttonSpeak.setOnClickListener(v -> {
String text = editText.getText().toString();
if (!text.isEmpty()) {
textToSpeechHelper.speakText(text);
} else {
```

```
Toast.makeText(MainActivity.this, "Please enter text to
speak", Toast.LENGTH_SHORT).show();} });
buttonListen.setOnClickListener(v -> {
speechToTextHelper.startSpeechRecognition());});
buttonClear.setOnClickListener(v -> {
editText.setText(""); // Clear the EditText
Toast.makeText(MainActivity.this, "Text cleared",
Toast.LENGTH_SHORT).show();});}
@Override ,protected void onDestroy() {
super.onDestroy();
textToSpeechHelper.shutdown();
speechToTextHelper.shutdown(); ; }
```

SpeechToTextHelper

```
public class SpeechToTextHelper implements
RecognitionListener {
private SpeechRecognizer speechRecognizer;
private OnSpeechRecognitionListener listener;
public void initSpeechToText(Context context) {
speechRecognizer =
SpeechRecognizer.createSpeechRecognizer(context);
speechRecognizer.setRecognitionListener(this); }
public void startSpeechRecognition() {
Intent intent = new
Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEE
CH);
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE
_MODEL,
RecognizerIntent.LANGUAGE_MODEL_FREE_FORM
);
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE,
Locale.getDefault());
speechRecognizer.startListening(intent); }
public void
setOnSpeechRecognitionListener(OnSpeechRecognition
Listener listener) {
this.listener = listener;
@Override public void onResults(Bundle results) {
ArrayList<String> recognizedText =
results.getStringArrayList(SpeechRecognizer.RESULTS
_RECOGNITION);
if (listener != null && recognizedText != null &&
!recognizedText.isEmpty()) {
listener.onSpeechRecognized(recognizedText.get(0)); } }
}
```

TextToSpeechHelper

```
package com.example.exe05;
import android.content.Context;
import android.speech.tts.TextToSpeech;
import android.util.Log;
import java.util.Locale;
public class TextToSpeechHelper {
private TextToSpeech textToSpeech;
public void initTextToSpeech(Context context) {
textToSpeech = new TextToSpeech(context, status -> {
if (status == TextToSpeech.SUCCESS) { int result =
textToSpeech.setLanguage(Locale.getDefault());
if (result == TextToSpeech.LANG_MISSING_DATA ||
result == TextToSpeech.LANG_NOT_SUPPORTED) {
Log.e("TTS", "Language not supported"); }
} else {
```

```

Log.e("TTS", "Initialization failed");}});}
public void speakText(String text) {
if (text != null && !text.isEmpty()) {
Log.d("TTS", "Speaking text: " + text);
textToSpeech.speak(text,
TextToSpeech.QUEUE_FLUSH, null, null);
} else {
Log.e("TTS", "Text is empty or null"); } }
public void shutdown() {
if (textToSpeech != null) {
textToSpeech.stop();
textToSpeech.shutdown();}} }

```

Activity_Main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@color/colorBackground">
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical"
android:padding="16dp">

<!-- Caption TextView -->
<TextView
android:id="@+id/captionTextView"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="MAD_exe05"
android:textSize="24sp"
android:textColor="@color/colorPrimary"
android:textStyle="bold"
android:gravity="center"
android:layout_marginBottom="16dp" />

<EditText
android:id="@+id/editText"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:hint="Enter text here"
style="@style/EditTextStyle" />

```

```

<!-- Horizontal LinearLayout for buttons -->
<LinearLayout
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="horizontal"
android:gravity="center"
android:layout_marginTop="16dp">

```

```

<Button
android:id="@+id/buttonSpeak"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Speak"
style="@style/SpeakButtonStyle"
android:layout_weight="1" />

```

```

<Button
android:id="@+id/buttonListen"
android:layout_width="0dp"
android:layout_height="wrap_content"
android:text="Listen"
style="@style/ListenButtonStyle"
android:layout_weight="1" />
</LinearLayout>

```

```

<Button
android:id="@+id/buttonClear"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:text="Clear"
style="@style/ClearButtonStyle"
android:layout_marginTop="16dp" />

```

```

<TextView
android:id="@+id/footer_text"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="BY YOGESWARAN S"
android:textSize="30sp"
android:textColor="@android:color/black"
android:fontFamily="cursive"
android:layout_gravity="bottom|end"
android:layout_marginBottom="32dp"
android:layout_marginEnd="16dp" />
</LinearLayout>
</ScrollView>

```

7:26

MAD_exe06

221501511

Speak

Listen

Clear

BY YOGESWARAN S