PRACTICAL: 12

AIM: Create an application to handle support voice interaction.

THEORY: Voice actions are an important part of the wearable experience. They let users carry out actions hands-free and quickly. Wear OS by Google provides two types of voice actions:

System-provided: These voice actions are task-based and are built into the Wear platform. You filter for them in the activity that you want to start when the voice action is spoken. Examples include "Take a note" or "Set an alarm".

App-provided: These voice actions are app-based, and you declare them just like a launcher icon. Users say "Start Your App Name" to use these voice actions and an activity that you specify starts.

CODE:

activity_main.xml

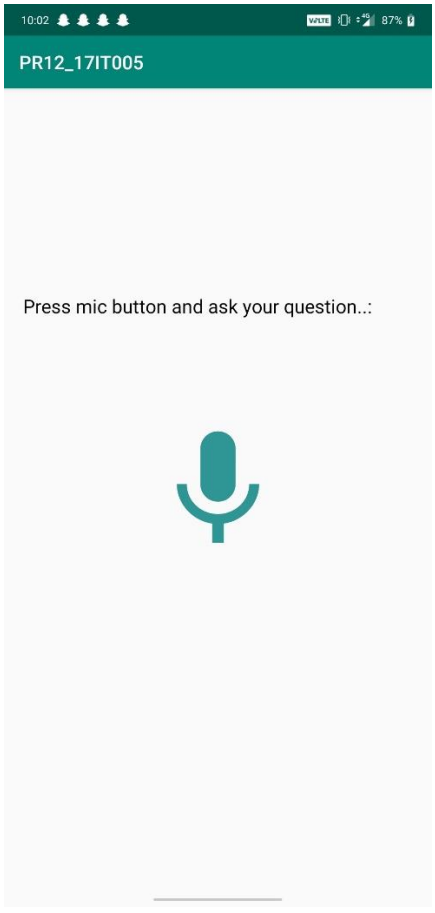
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textTv"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_above="@+id/voiceBtn"
        android:layout_marginBottom="97dp"
        android:text="Press mic button and ask your question..:"
        android:textColor="#000"
        android:paddingVertical="10dp"
        android:textSize="20dp" />

    <ImageButton
        android:id="@+id/voiceBtn"
        android:layout_width="150dp"
        android:layout_height="150dp"
        android:background="@null"
        android:src="@drawable/ic_mic"
        android:scaleType="fitCenter"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"/>

</RelativeLayout>
```

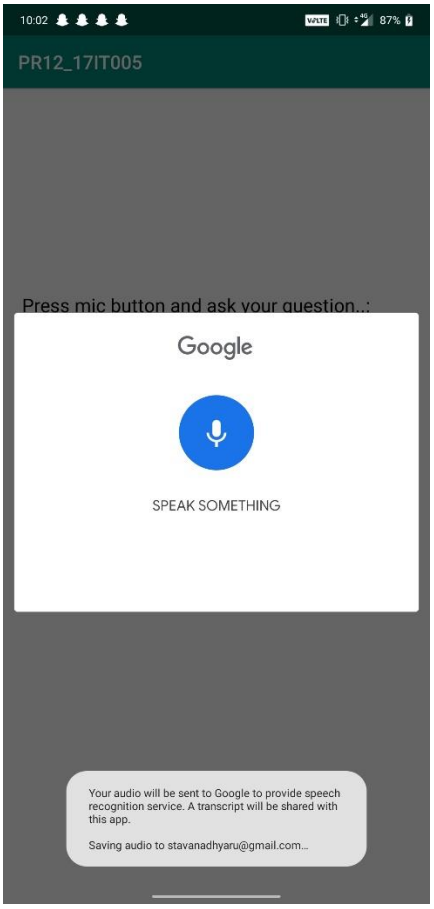
MainActivity.java

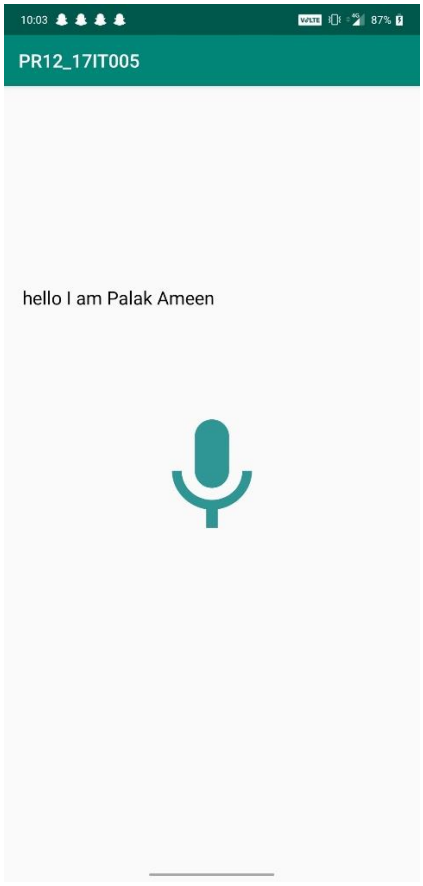
```
package com.example.pr12_17it005;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.speech.RecognizerIntent;
import android.view.View;
import android.widget.ImageButton;
import android.widget.TextView;
import android.widget.Toast;
import java.util.ArrayList;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {
    private static final int REQUEST_CODE_SPEECH_INPUT =1000 ;
    TextView mtexttv;
    ImageButton mVoicebtn;

   @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mtexttv=findViewById(R.id.textTv);
        mVoicebtn=findViewById(R.id.voiceBtn);
        mVoicebtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                speak();
            }
        });
    }

    private void speak() {
         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,"SPEAK SOMETHING");

            try {
                //in there was no error
                startActivityForResult(intent,REQUEST_CODE_SPEECH_INPUT);

            }
            catch (Exception e){
                //if error
                Toast.makeText(this,""+e.getMessage(),Toast.LENGTH_SHORT).show();
            }
    }
```

```
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
          switch (requestCode){
           case REQUEST_CODE_SPEECH_INPUT:{

               if (resultCode==RESULT_OK && null!=data){
                   ArrayList<String> result
=data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
               mtexttv.setText(result.get(0));

               }
                   break;
           }
       }
    }
}
```

OUTPUT:







Here, speech is recognized and shown as text on the screen.

LATEST APPLICATIONS: Voice recognition is mainly used by google, alexa, siri, and many more applications or products, it was first implemented in windows 98 where program was called cortona which is still used, and in ios it was first used in iphone 4S which created the boom for voice recognition in the market.

LEARNING OUTCOME: To implement voice support module of google in android application certain dependencies need to be added in android manifest file. We use Intent recognizer for speech recognition and store and display result with the help of ArrayList.