PRACTICAL: 12

AIM:

Create an application to handle support voice interaction.

THEORY:

A voice interaction is a special kind of Android activity that is triggered by the user's voice that also lets them complete an action by voice.

The application uses TextToSpeech instance, which synthesizes speech from text for immediate playback or to create a sound file. A TextToSpeech instance can only be used to synthesize text once it has completed its initialization. Implementation of the TextToSpeech.onInitListener is to get notified of the completion of the initialization. After completion of using the TextToSpeech instance, the shutdown() method is used to release the native resources used by the TextToSpeech engine.

In order to trigger API, one need to start an Intent which is android.speech.RecognizerIntent, which shows mic dialog box to recognize speech input. This Activity converts the speech into text and sends backs the result to our calling Activity. The speech recognizer intent is started using startActivityForResult() with bundled extras. When we invoke android.speech.RecognizerIntent intent, we must use startActivityForResult() as we must get back the result text.

CODE:

Program: activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    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"
    tools:context="com.example.henyd.prac12.MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="168dp"
        android:text="Wait till Question PopUP!!"
        android:textSize="24sp"
        app:layout_constraintHorizontal_bias="0.501"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />
```

```
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginEnd="8dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:text="Speak your answer"
        android:textSize="24sp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.501"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView"
        app:layout_constraintVertical_bias="0.171" />
</android.support.constraint.ConstraintLayout>
```

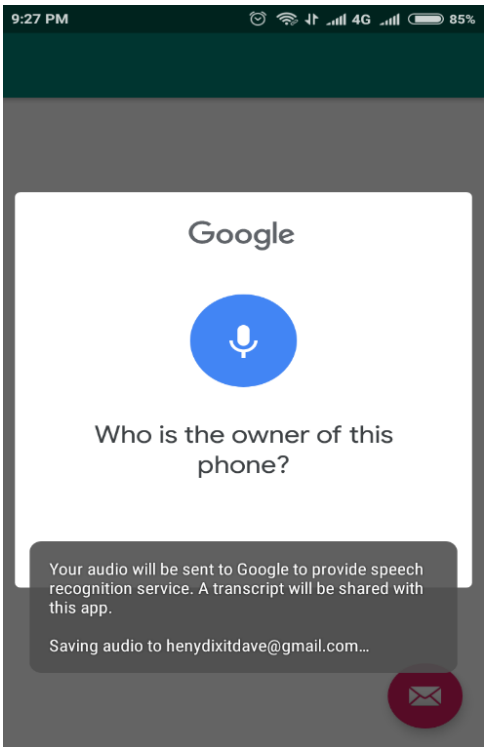
Program: MainActivity.java

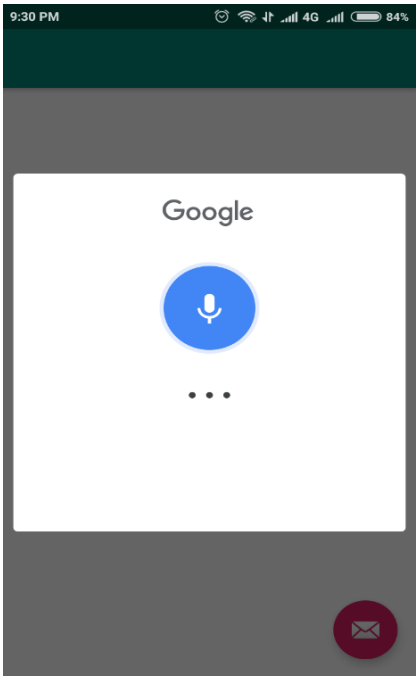
```
package com.example.henyd.prac12;
import android.content.Intent;
import android.speech.RecognizerIntent;
import android.speech.tts.TextToSpeech;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.TextView;
import java.util.List;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    private TextToSpeech t1;
    private final int REQUEST_SPEECH_RECOGNIZER = 3000;
    private TextView question, answer;
    private final String mQuestion = "Who is the owner of this phone?";
    private String mAnswer = "";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        question = (TextView) findViewById(R.id.textView);
        answer = (TextView) findViewById(R.id.textView1);
        t1=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {
```

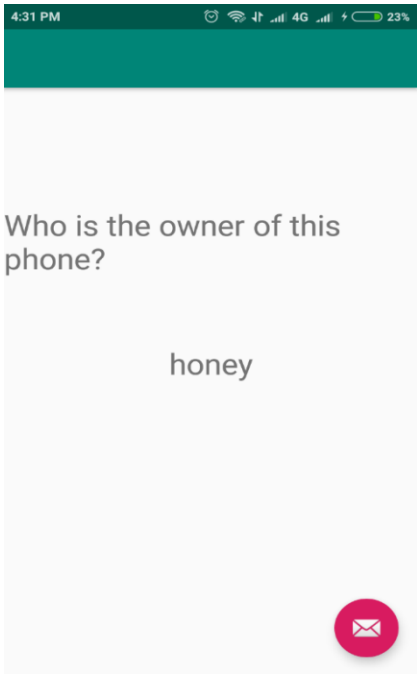
```
        @Override
        public void onInit(int status) {
            if(status != TextToSpeech.ERROR) {
                t1.setLanguage(Locale.UK);
            }
        }
    });
    startSpeechRecognizer();
  }
  private void startSpeechRecognizer() {
    Intent intent = new Intent
          (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
    intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
          RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
    intent.putExtra(RecognizerIntent.EXTRA_PROMPT, mQuestion);
    startActivityForResult(intent, REQUEST_SPEECH_RECOGNIZER);
  }
  @Override
  protected void onActivityResult(int requestCode, int resultCode,
                        Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == REQUEST_SPEECH_RECOGNIZER) {
      if (resultCode == RESULT_OK) {
        List<String> results = data.getStringArrayListExtra
              (RecognizerIntent.EXTRA_RESULTS);
        mAnswer = results.get(0);
        question.setText(mQuestion);
        answer.setText(mAnswer);
        if (mAnswer.toUpperCase().indexOf("SMIT") > -1) {
          t1.speak("Great You are correct", TextToSpeech.QUEUE_FLUSH, null,
"adfvsfgbrsgh");
        }
        else {
          t1.speak("Wrong answer submit this phone to my owner Smit",
TextToSpeech.QUEUE_FLUSH, null, "adfvsfgbrsgh");
        }
      }
    }
  }
  @Override
```

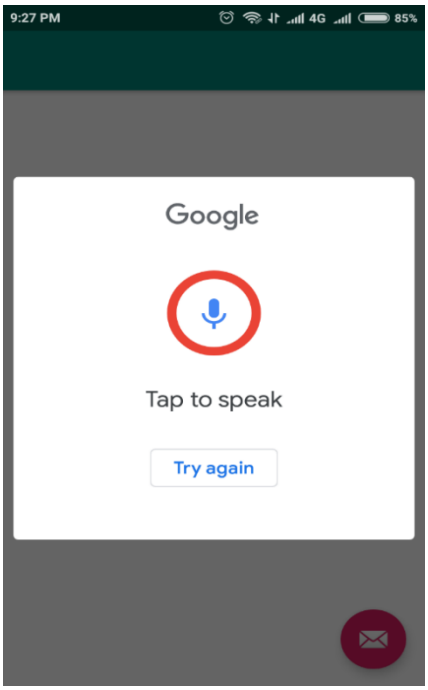
```
    public void onPause(){
        if(t1 !=null){
            t1.stop();
            t1.shutdown();
        }
        super.onPause();
    }
}
```

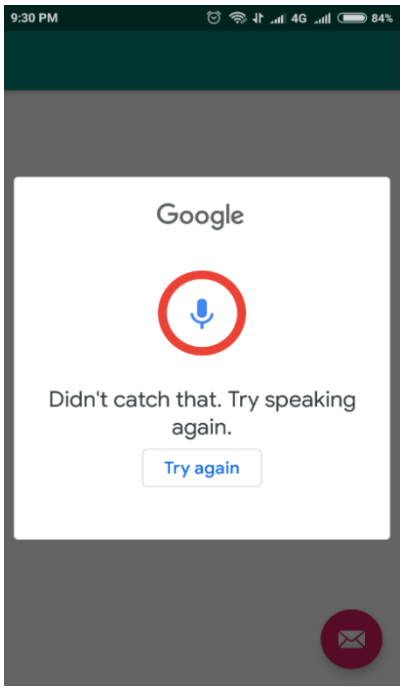
OUTPUT:











CONCLUSION:

We have successfully created android application in which recognizes speech and stores it and displays it on the home screen of the application. Thus, the application handles human voice interaction.