*DOCUMENTATION*

Project: - Text To Speech App using Android

Group Member:-

1. Aashutosh Tewari, MCKV Institute of Technology

2. Pradip Bewra, Narula Institute of Technology

3. Shrinjoy Guha, Narula Institute of Technology

4. Subhajit Pal, Narula Institute of Technology

5. Subhanjan Neogi,Narula Institute of Technology

***ACKNOWLODGEMENT:-***

I take this opportunity to express my profound gratitude and deep to my faculty (Sir Varun Gupta) for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by him/her time to shall carry me a long way in the journey of life on which I am about embark.

I am obliged to my project team members for the valuable information provide by them in their respected field. I am grateful for their cooperation during the period of my assignment.

***Flow Diagram:-***

Main Activity

1st layout

(Welcome Page)

TTS Activity

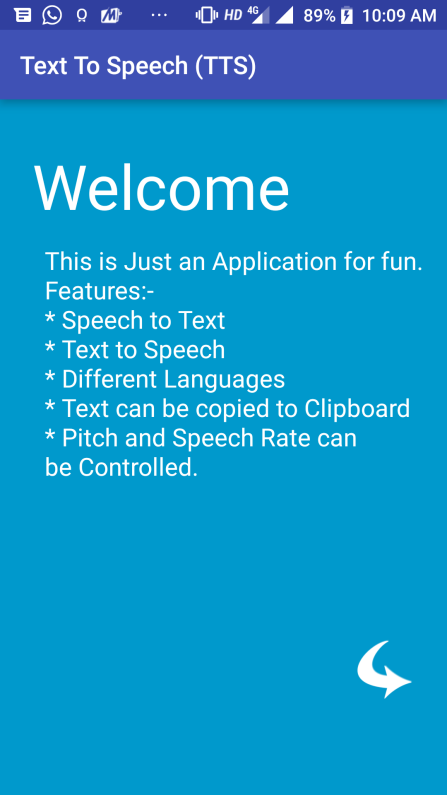
2nd Layout

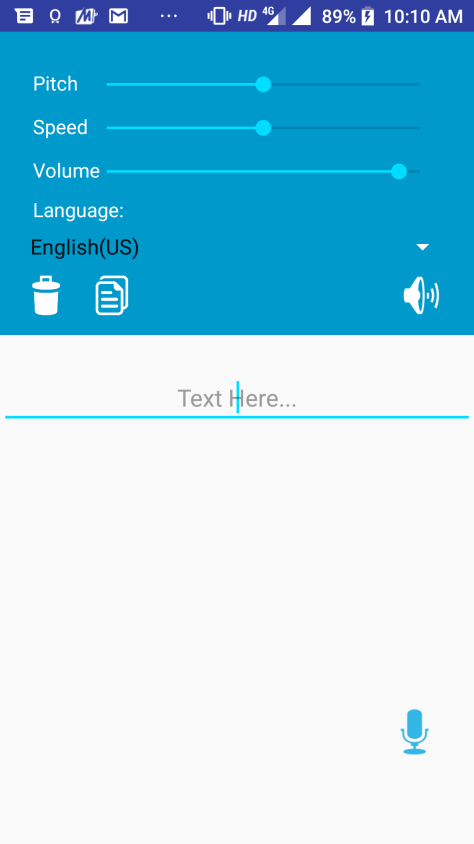
(Voice Modulator)

Quit the App manually

*Flow diagram of TTS App*

***Screenshot:-***

****

****

***SOURCE CODE:-***

**Layout1:-**

<?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"  
 xmlns:ads="http://schemas.android.com/apk/res-auto"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context="souravapp.tts.MainActivity"  
 android:background="@android:color/holo\_blue\_dark">  
   
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginTop="20dp"  
 android:layout\_marginLeft="10dp"  
 android:textColor="@android:color/white"  
 android:textSize="50sp"  
 android:text="Welcome"  
 tools:ignore="HardcodedText,RtlHardcoded" />  
 <TextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:layout\_alignParentTop="true"  
 android:layout\_marginTop="100dp"  
 android:id="@+id/intro"  
 android:layout\_marginLeft="20dp"  
 android:textColor="@android:color/white"  
 android:textSize="20sp"  
 android:text="This is Just an Application for fun.  
 \nFeatures:-  
 \n\* Speech to Text  
 \n\* Text to Speech  
 \n\* Different Languages  
 \n\* Text can be copied to Clipboard  
 \n\* Pitch and Speech Rate can be Controlled."  
 tools:ignore="HardcodedText,RtlHardcoded" />  
  
 <Button  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/arrow"  
 android:backgroundTint="@android:color/white"  
 android:id="@+id/next\_page"  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginBottom="60dp"  
 android:layout\_marginRight="10dp"  
 tools:targetApi="lollipop"  
 tools:ignore="RtlHardcoded" />  
</RelativeLayout>

**MainActivity:-**

package souravapp.tts;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.support.v7.app.AppCompatActivity;  
import android.view.View;  
import android.view.animation.Animation;  
import android.view.animation.AnimationUtils;  
import android.widget.Button;  
import android.widget.TextView;  
  
import com.google.android.gms.ads.AdRequest;  
import com.google.android.gms.ads.AdView;  
  
public class MainActivity extends AppCompatActivity {  
 Button nxtpg;  
 TextView intro;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
  
 intro = (TextView) findViewById(R.id.*intro*);  
 Animation animation =  
 AnimationUtils.*loadAnimation*(getApplicationContext(), R.anim.*movetxt*);  
 intro.startAnimation(animation);  
 nxtpg = (Button) findViewById(R.id.*next\_page*);  
 nxtpg.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent i;  
 i = new Intent(MainActivity.this, TTS.class);  
 startActivity(i);  
 }  
 });  
 }  
  
}

**Layout 2:-**

<?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"  
 xmlns:ads="http://schemas.android.com/apk/res-auto">  
 <RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="230dp"  
 android:padding="15dp"  
 android:background="@android:color/holo\_blue\_dark"  
 android:id="@+id/option">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Language:"  
 android:textSize="15sp"  
 android:layout\_alignParentBottom="true"  
 android:layout\_marginBottom="70dp"  
 android:layout\_marginLeft="10dp"  
 android:textColor="@android:color/white"  
 tools:ignore="HardcodedText,RtlHardcoded" />  
  
 <Spinner  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:backgroundTint="@android:color/white"  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginBottom="40dp"  
 android:id="@+id/spinner"  
 tools:targetApi="lollipop" />  
   
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 android:text="Volume"  
 android:textSize="15sp"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginBottom="100dp"  
 android:textColor="@android:color/white"  
 tools:ignore="HardcodedText,RelativeOverlap,RtlHardcoded" />  
 <SeekBar  
 android:layout\_width="270dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentBottom="true"  
 android:layout\_marginBottom="100dp"  
 android:layout\_marginRight="10dp"  
 android:id="@+id/volumeBar"  
 tools:ignore="RtlHardcoded" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:text="@string/speed"  
 android:layout\_marginBottom="15dp"  
 android:layout\_above="@id/volumeBar"  
 android:textSize="15sp"  
 android:layout\_marginLeft="10dp"  
 android:textColor="@android:color/white"  
 tools:ignore="RelativeOverlap,RtlHardcoded" />  
  
 <SeekBar  
 android:layout\_width="270dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="15dp"  
 android:layout\_above="@id/volumeBar"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginRight="10dp"  
 android:id="@+id/speedBar"  
 tools:ignore="RtlHardcoded" />  
  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentStart="true"  
 android:text="Pitch"  
 android:layout\_marginBottom="15dp"  
 android:layout\_above="@id/speedBar"  
 android:textSize="15sp"  
 android:layout\_marginLeft="10dp"  
 android:textColor="@android:color/white"  
 tools:ignore="HardcodedText,RelativeOverlap,RtlHardcoded" />  
  
 <SeekBar  
 android:layout\_width="270dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="15dp"  
 android:layout\_above="@id/speedBar"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginRight="10dp"  
 android:id="@+id/pitchBar"  
 tools:ignore="RtlHardcoded" />  
 </RelativeLayout>  
  
 <EditText  
 android:layout\_width="400dp"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText"  
 android:gravity="center"  
 android:hint="Text Here..."  
 android:layout\_below="@id/option"  
 android:layout\_marginTop="25dp"  
 android:layout\_marginBottom="20dp"  
 android:layout\_centerVertical="true"  
 android:layout\_centerHorizontal="true"  
 tools:ignore="HardcodedText" />  
 <Button  
 android:layout\_width="40dp"  
 android:layout\_height="40dp"  
 android:background="@android:drawable/ic\_lock\_silent\_mode\_off"  
 android:backgroundTint="@android:color/white"  
 android:id="@+id/spk"  
 android:layout\_gravity="center\_vertical"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginTop="180dp"  
 tools:targetApi="lollipop" />  
 <Button  
 android:layout\_width="30dp"  
 android:layout\_height="30dp"  
 android:background="@drawable/delete"  
 android:backgroundTint="@android:color/white"  
 android:id="@+id/del"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginStart="20dp"  
 android:layout\_marginTop="185dp"  
 tools:targetApi="lollipop" />  
 <Button  
 android:layout\_width="30dp"  
 android:layout\_height="30dp"  
 android:background="@drawable/copy"  
 android:backgroundTint="@android:color/white"  
 android:id="@+id/copy"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginStart="70dp"  
 android:layout\_marginTop="185dp"  
 tools:targetApi="lollipop" />  
  
 <Button  
 android:layout\_width="50dp"  
 android:layout\_height="50dp"  
 android:background="@android:drawable/ic\_btn\_speak\_now"  
 android:backgroundTint="@android:color/holo\_blue\_light"  
 android:id="@+id/speech\_to\_text"  
 android:layout\_marginTop="10dp"  
 android:layout\_marginLeft="10dp"  
 android:layout\_alignParentBottom="true"  
 android:layout\_alignParentEnd="true"  
 android:layout\_marginEnd="20dp"  
 android:layout\_marginBottom="60dp"  
 tools:targetApi="lollipop"  
 tools:ignore="RtlHardcoded" />  
  
</RelativeLayout>

**TTS.java:-**

package souravapp.tts;  
  
import android.app.Activity;  
import android.content.ActivityNotFoundException;  
import android.content.ClipData;  
import android.content.ClipboardManager;  
import android.content.Context;  
import android.content.Intent;  
import android.media.AudioManager;  
import android.os.Bundle;  
import android.speech.RecognizerIntent;  
import android.speech.tts.TextToSpeech;  
import android.view.View;  
import android.widget.ArrayAdapter;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.SeekBar;  
import android.widget.Spinner;  
import android.widget.Toast;  
  
import com.google.android.gms.ads.AdListener;  
import com.google.android.gms.ads.AdRequest;  
import com.google.android.gms.ads.AdView;  
import com.google.android.gms.ads.InterstitialAd;  
  
import java.util.ArrayList;  
import java.util.Locale;  
  
*/\*\*  
 \* Created by SOURAV on 1/27/2017.  
 \*/*public class TTS extends Activity {  
 TextToSpeech tts;  
 EditText text;  
 Button spk,del,cpy,mic;  
 Spinner Language\_Select;  
 int Language;  
 SeekBar Pitch,Volume,Speed;  
 AudioManager audioManager;  
 private final int SPEECH\_RECOGNITION\_CODE = 1;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*tts*);  
  
  
  
  
  
 text=(EditText)findViewById(R.id.*editText*);  
  
 spk=(Button)findViewById(R.id.*spk*);  
 del=(Button)findViewById(R.id.*del*);  
 cpy=(Button)findViewById(R.id.*copy*);  
 mic=(Button)findViewById(R.id.*speech\_to\_text*);  
  
  
 Language\_Select=(Spinner)findViewById(R.id.*spinner*);  
 ArrayAdapter<CharSequence> adapter=ArrayAdapter.*createFromResource*(this,R.array.*languages*,android.R.layout.*simple\_spinner\_item*);  
 adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*);  
 Language\_Select.setAdapter(adapter);  
  
 Pitch=(SeekBar)findViewById(R.id.*pitchBar*);  
 Pitch.setProgress(50);  
  
 Speed=(SeekBar)findViewById(R.id.*speedBar*);  
 Speed.setProgress(50);  
  
 Volume=(SeekBar)findViewById(R.id.*volumeBar*);  
  
  
 audioManager =(AudioManager)getSystemService(Context.*AUDIO\_SERVICE*);  
 Volume.setMax(audioManager.getStreamMaxVolume(AudioManager.*STREAM\_MUSIC*));  
 Volume.setProgress(audioManager.getStreamVolume(AudioManager.*STREAM\_MUSIC*));  
  
 Pitch.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
  
 double pitch =Pitch.getProgress()+1;  
 pitch= pitch/25;  
 tts.setPitch((float)pitch);  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
 });  
  
 Speed.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
  
 double speed =Speed.getProgress()+1;  
 speed= speed/30;  
 tts.setSpeechRate((float)speed);  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
 });  
  
 Volume.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {  
 @Override  
 public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {  
 audioManager.setStreamVolume(AudioManager.*STREAM\_MUSIC*,(progress),0);  
 }  
  
 @Override  
 public void onStartTrackingTouch(SeekBar seekBar) {  
  
 }  
  
 @Override  
 public void onStopTrackingTouch(SeekBar seekBar) {  
  
 }  
 });  
  
 mic.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 startSpeechToText();  
 }  
 });  
  
 tts =new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {  
 @Override  
 public void onInit(int status) {  
 if(status!=tts.*ERROR*){  
 tts.setLanguage(Locale.*ENGLISH*);  
 }  
 }  
 });  
  
 del.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 if(text.getText().toString().length()==0) {  
 Toast.*makeText*(getApplicationContext(), "Nothing to Delete!!!", Toast.*LENGTH\_SHORT*).show();  
 }  
 else {  
 text.setText("");  
 Toast.*makeText*(getApplicationContext(), "Text Deleted!!!", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
  
 spk.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 if(text.getText().toString().length()==0) {  
 Toast.*makeText*(getApplicationContext(), "Nothing to Speak!!!", Toast.*LENGTH\_SHORT*).show();  
 }  
 else {  
 String tospeak = text.getText().toString();  
  
 switch (Language\_Select.getSelectedItemPosition()){  
 case 0:  
 tts.setLanguage(Locale.*ENGLISH*);  
 break;  
 case 1:  
 tts.setLanguage(Locale.*UK*);  
 break;  
 case 2:  
 tts.setLanguage(new Locale("es","ES"));  
 break;  
 case 3:  
 tts.setLanguage(Locale.*FRENCH*);  
 break;  
 case 4:  
 tts.setLanguage(Locale.*GERMAN*);  
 break;  
 case 5:  
 tts.setLanguage(Locale.*ITALIAN*);  
 break;  
 case 6:  
 tts.setLanguage(new Locale("ru","RU"));  
 break;  
 case 7:  
 tts.setLanguage(new Locale("hi","IN"));  
 break;  
  
 default:  
 tts.setLanguage(Locale.*ENGLISH*);  
 break;  
 }  
 tts.speak(tospeak, TextToSpeech.*QUEUE\_FLUSH*, null);  
 }  
 }  
 });  
  
 cpy.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 if(text.getText().toString().length()==0) {  
 Toast.*makeText*(getApplicationContext(), "Nothing to Copy!!!", Toast.*LENGTH\_SHORT*).show();  
 }  
 else {  
 ClipboardManager clipbrd = (ClipboardManager)getSystemService(*CLIPBOARD\_SERVICE*);  
 ClipData clip =ClipData.*newPlainText*("TTS text",text.getText().toString());  
 clipbrd.setPrimaryClip(clip);  
 Toast.*makeText*(getApplicationContext(), "Text Copied to Clipboard!!!", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 });  
 }  
 @Override  
 protected void onPause() {  
 super.onPause();  
 if(tts !=null){  
 tts.stop();  
 }  
 }  
  
 @Override  
 protected void onDestroy() {  
  
 if(tts !=null){  
 tts.stop();  
 tts.shutdown();  
 }  
 super.onDestroy();  
 }  
  
 */\*\*  
 \* Start speech to text intent. This opens up Google Speech Recognition API dialog box to listen the speech input.  
 \* \*/* private void startSpeechToText() {  
 Intent intent = new Intent(RecognizerIntent.*ACTION\_RECOGNIZE\_SPEECH*);  
 intent.putExtra(RecognizerIntent.*EXTRA\_LANGUAGE*, Locale.*getDefault*());  
 intent.putExtra(RecognizerIntent.*EXTRA\_LANGUAGE\_MODEL*,  
 RecognizerIntent.*LANGUAGE\_MODEL\_FREE\_FORM*);  
 intent.putExtra(RecognizerIntent.*EXTRA\_PROMPT*,  
 "Speak something...");  
 try {  
 startActivityForResult(intent, SPEECH\_RECOGNITION\_CODE);  
 } catch (ActivityNotFoundException a) {  
 Toast.*makeText*(getApplicationContext(),  
 "Sorry! Speech recognition is not supported in this device.",  
 Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 */\*\*  
 \* Callback for speech recognition activity  
 \* \*/* @Override  
 protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
 switch (requestCode) {  
 case SPEECH\_RECOGNITION\_CODE: {  
 if (resultCode == *RESULT\_OK* && null != data) {  
 ArrayList<String> result = data  
 .getStringArrayListExtra(RecognizerIntent.*EXTRA\_RESULTS*);  
 text.setText(text.getText().toString()+" "+result.get(0));  
 }  
 break;  
 }  
 }  
 }  
}