Demo33-VideoPlayer视频播放

使用VideoView控件来实现

1. activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" >

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" >

<Button

android:id="@+id/play"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Play" />

<Button

android:id="@+id/pause"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Pause" />

<Button

android:id="@+id/replay"

android:layout\_width="0dp"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:text="Replay" />

</LinearLayout>

<VideoView

android:id="@+id/video\_view"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

</LinearLayout>

1. MainActivity.java

import android.Manifest;

import android.content.pm.PackageManager;

import android.net.Uri;

import android.os.Environment;

import android.support.v4.app.ActivityCompat;

import android.support.v4.content.ContextCompat;

import android.support.v7.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

import android.widget.VideoView;

import java.io.File;

public class MainActivity extends AppCompatActivity implements View.OnClickListener{

private VideoView videoView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

videoView = (VideoView) findViewById(R.id.video\_view);

Button play = (Button) findViewById(R.id.play);

Button pause = (Button) findViewById(R.id.pause);

Button replay = (Button) findViewById(R.id.replay);

play.setOnClickListener(this);

pause.setOnClickListener(this);

replay.setOnClickListener(this);

if (ContextCompat.checkSelfPermission(MainActivity.this, Manifest.permission.WRITE\_EXTERNAL\_STORAGE) != PackageManager.PERMISSION\_GRANTED) {

ActivityCompat.requestPermissions(MainActivity.this, new String[]{ Manifest.permission. WRITE\_EXTERNAL\_STORAGE }, 1);

} else {

initVideoPath(); // 初始化MediaPlayer

}

}

private void initVideoPath() {

File file = new File(Environment.getExternalStorageDirectory(), "movie.mp4");

videoView.setVideoPath(file.getPath()); // 指定视频文件的路径

}

@Override

public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {

switch (requestCode) {

case 1:

if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION\_GRANTED) {

initVideoPath();

} else {

Toast.makeText(this, "拒绝权限将无法使用程序", Toast.LENGTH\_SHORT).show();

finish();

}

break;

default:

}

}

@Override

public void onClick(View v) {

switch (v.getId()) {

case R.id.play:

if (!videoView.isPlaying()) {

videoView.start(); // 开始播放

}

break;

case R.id.pause:

if (videoView.isPlaying()) {

videoView.pause(); // 暂停播放

}

break;

case R.id.replay:

if (videoView.isPlaying()) {

videoView.resume(); // 重新播放

}

break;

}

}

@Override

protected void onDestroy() {

super.onDestroy();

if (videoView != null) {

videoView.suspend();

}

}

}

3、AndroidManifest.xml添加权限

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />