

行動裝置數位匯流應用實務

錄音程式

王昱景 Brian Wang

brian.wang.frontline@gmail.com

- 一般來說，行動裝置的麥克風除了通話外，也可以建立錄音程式
- 使用麥克風裝置來錄製聲音，在Android是使用MediaRecorder物件來錄音和將它存入檔案
- MediaRecorder物件在使用上只需要指定錄音格式和儲存檔案的路徑，就可以開始進行聲音錄製

RecorderDemo

- 開啟和執行Android專案
- 建立錄音程式的版面配置
- 建立Activity活動類別的事件處理方法
- 在AndroidManifest.xml新增錄音權限

I. 開啟和執行Android專案

- 請啟動 Eclipse IDE
- 建立 Android 專案
 - Project Name: RecordDemo
 - Build Target: Android 4.2.2
 - Package Name

2. 建立錄音程式的版面配置

- 錄音程式的使用介面是定義在 `activity_main.xml` 版面配置檔
- 使用 `LinearLayout` 垂直編排 1 個 `TextView` 和 2 個 `Button` 元件

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/LinearLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >

    <TextView
        android:id="@+id/output"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/label" />

    <Button
        android:id="@+id/startBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/start"
        android:enabled="false" />

    <Button
        android:id="@+id/stopBtn"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="@string/stop"
        android:enabled="false" />

</LinearLayout>
```



3. 建立Activity活動類別的事件處理方法

- 在活動類別的開頭宣告成員的
MediaRecorder、Button、TextView和File
物件變數

```
public class MainActivity extends Activity {  
  
    private MediaRecorder recorder;  
    private Button startBtn, stopBtn;  
    private TextView output;  
    private File path;  
  
}
```

onCreate() 方法

- 在覆寫的onCreate()方法建立MediaRecorder物件和錄音檔路徑的File物件
- 取得Button和TextView元件後，建立MediaRecorder物件
- 在取得SD卡外部儲存裝置的路徑來建立錄音檔的File物件後，呼叫之後的resetRecorder()方法重設MediaRecorder物件


```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // 取得Button元件
    startBtn = (Button) findViewById(R.id.startBtn);
    startBtn.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View v) {
            output.setText("開始錄音....");

            try {
                recorder.start(); // 開始錄音

                // 設定按鈕狀態
                startBtn.setEnabled(false);
                stopBtn.setEnabled(true);
            } catch (Exception ex) {
                Log.d("RecordDemo", "startBtnClick: " + ex.getMessage());
            }
        }
    });

    stopBtn = (Button) findViewById(R.id.stopBtn);
    stopBtn.setOnClickListener(new OnClickListener() {

        @Override
        public void onClick(View v) {
            output.setText("停止錄音....");

            recorder.stop(); // 停止錄音
            // 設定按鈕狀態
            startBtn.setEnabled(true);
            stopBtn.setEnabled(false);
        }
    });

    output = (TextView) findViewById(R.id.output);
    recorder = new MediaRecorder();
    path = new File(Environment.getExternalStorageDirectory(), "myrecord.3gp");

    resetRecorder();
}

```

onDestroy()方法

- 在覆寫onDestroy()方法呼叫release()方法釋放MediaRecorder物件的資源

```
@Override  
protected void onDestroy() {  
    super.onDestroy();  
    recorder.release();  
}
```

resetRecorder()方法

- 在自定的resetRecorder()方法重設錄音機的MediaRecorder物件

```
// 重設錄音機
private void resetRecorder() {
    recorder.setAudioSource(MediaRecorder.AudioSource.MIC);
    recorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
    recorder.setAudioEncoder(MediaRecorder.AudioEncoder.DEFAULT);
    recorder.setOutputFile(path.getAbsolutePath());

    try {
        recorder.prepare();
        output.setText("錄音程序準備完成....");
        startBtn.setEnabled(true);
    } catch (Exception ex) {
        Log.d("RecordDemo", "resetRecorder: " + ex.getMessage());
    }
}
```

方法	說明
<code>setAudioSource()</code>	指定錄音來源， <code>AudioSource.MIC</code> 是麥克風，此方法需在 <code>setOutputFormat()</code> 前呼叫
<code>setOutputFormat()</code>	指定輸出錄音檔的格式， <code>OutputFormat.THREE_GPP</code> 是3GPP格式； <code>OutputFormat.MPEG_4</code> 是MP4格式； <code>OutputFormat.RAW_AMR</code> 是AMR NB格式
<code>setAudioEncoder()</code>	指定錄音使用的編碼器， <code>AudioEncoder.DEFAULT</code> 是預設編碼器
<code>setOutputFile()</code>	指定輸出檔案的路徑是參數字串
<code>prepare()</code>	<code>MediaRecorder</code> 物件已經準備好可以開始錄音

4. 在AndroidManifest.xml新增錄音權限

- 錄音程式因為執行錄音和在外部儲存裝置寫入檔案，所以在AndroidManifest.xml檔案需要新增**RECORD_AUDIO**和**WRITE_EXTERNAL_STORAGE**權限

```
<uses-permission android:name="android.permission.RECORD_AUDIO"/>  
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
```