

Hướng dẫn lập trình cơ bản với Android

Part 7

Trong bài 1 mình đã giới thiệu sơ lược về các thành phần cơ bản của Android cũng như việc sử dụng XML để lập trình ứng dụng Android. Trong bài này mình sẽ giới thiệu thêm về Android Manifest và đi sâu hơn về vấn đề làm việc với View.

Android Manifest

Trong khung Package Explorer, ở phía dưới thư mục res, bạn sẽ thấy 1 file có tên là AndroidManifest.xml. Mỗi ứng dụng đều cần có AndroidManifest.xml để mô tả những thông tin quan trọng của nó cho hệ thống Android biết. Let's look closer:
Mã:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="at.exam"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon"
android:label="@string/app_name">
        <activity android:name=".Example"
            android:label="@string/app_name">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
    <uses-sdk android:minSdkVersion="3" />
</manifest>
```

Cụ thể những công việc mà AndroidManifest.xml thực hiện:

- Đặt tên cho Java package của ứng dụng.
- Mô tả các thành phần (component) của ứng dụng: activity, service, broadcast receiver hoặc content provider.
- Thông báo những permission mà ứng dụng cần có để truy nhập các protected API và tương tác với các ứng dụng khác.
- Thông báo những permission mà các ứng dụng khác cần có để tương tác với ứng dụng hiện thời.
- Thông báo level thấp nhất của Android API mà ứng dụng cần để chạy. (Android 1.0 là level 1, 1.1 là level 2, 1.5 level 3, 1.6 level 4 và 2.0 là level 5).
- ...

Hãy xem thử file AndroidManifest.xml của chương trình ToDo mình đang xây dựng:

Mã:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="android.at"
    android:versionCode="1"
    android:versionName="1.0">
    <application android:icon="@drawable/icon"
android:label="@string/app_name">
        <activity
            android:name=".ToDo"
            android:screenOrientation="landscape"
            android:theme="@android:style/Theme.NoTitleBar.Fullscreen"
            android:label="@string/app_name">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
```

```
        </intent-filter>
    </activity>
    <activity android:name=".WorkEnter">
    </activity>
    <receiver android:name=".AlarmReceiver">
    </receiver>
</application>
<uses-sdk android:minSdkVersion="3" />
    <uses-permission
android:name="android.permission.VIBRATE"/>
</manifest>
```

Main Activity của chương trình Too Do này là activity TooDo. Ngoài ra mình còn có 1 Activity khác có tên là WorkEnter để cho phép nhập vào thời gian và nội dung công việc. 1 Broadcast Receiver có tên là AlarmReceiver để nhận alarm gửi tới trong intent.

Khi alarm được nhận sẽ có âm thanh và rung (vibration). Tất cả công việc sẽ được viết trong code, nhưng bắt buộc bạn phải khai báo các thành phần có trong ứng dụng vào AndroidManifest nếu muốn chương trình hoạt động.

Tương tự, set permission để truy nhập camera, internet, đọc contact... cũng đều phải khai báo trong AM. Từ khóa **screenOrientation** cho phép thiết lập giao diện khi vào ứng dụng theo chiều dọc (portrait - mặc định) hay ngang (landscape), **theme** cho phép sử dụng style có sẵn của android là full-screen (không có thanh status bar nữa).

Intent filter là bộ lọc dùng để giới hạn các intent được sử dụng trong activity hay receiver...

Mã:

```
<intent-filter>
    <action
android:name="android.intent.action.VIEW"/>
    <category
android:name="android.intent.category.DEFAULT"/>
    <category
android:name="android.intent.category.BROWSABLE"/>
```

```
<data android:scheme="http"
android:host="www.google.com"
android:path="/m/products/scan"/>
</intent-filter>
```

Bộ lọc trên chỉ cho phép intent mở internet với đường dẫn định nghĩa sẵn

(<http://www.google.com/m/products/scan>)

Ok, hi vọng mọi người đã nắm được chức năng cơ bản cũng như cách sử dụng Android Manifest

Working with View

Trong bài 1 mình đã giới thiệu qua cách sử dụng Edit Text và Text View. Thực chất các View còn lại cũng có cách sử dụng tương tự, bạn sẽ kết hợp nhiều View khác nhau để cho ra giao diện mình mong muốn. Ở đây mình sẽ đề cập nhiều tới List View (theo ý kiến mình là View khó sử dụng nhất).

Yêu cầu: Xây dựng một chương trình cho phép nhập nội dung công việc và thời gian rồi list ra

BI: Vẫn bắt đầu bằng cách khởi tạo một Project mới: File -> New -> Android Project.

Project name: Example 2

Build Target: Chọn Android 1.5

Application name: Example 2

Package name: at.exam

Create Activity: Example

=> Kích nút Finish.

Giao diện ta thiết kế ở đây có 1 Linear Layout làm thành phần chính, các thành phần con của nó gồm 1 Edit Text (dùng để nhập nội dung công việc), 1 Linear Layout (lại gồm các thành phần con để nhập giờ và phút thực hiện công việc), 1 Button (để thêm nội dung công việc vào List View) và 1 List View dùng để list các công việc bạn đã nhập.

Từ khóa **lines** được dùng để cố định số dòng và nên sử dụng với Edit Text thay vì dùng mỗi wrap_content vì nếu sd wrap_content thì Edit Text sẽ tự giãn ra nếu dòng nhập vào vượt giới hạn đường bao (làm hỏng giao diện bạn thiết kế).

Từ khóa **gravity** thông báo các thành phần con sẽ được sắp xếp ntn ở thành phần

cha. Ở đây mình dùng "center" nghĩa là thành phần con nằm ở trung tâm. Hãy thử thêm vào 1 Edit Text:

Mã:

```
android:gravity="center"
```