

Taller de Desarrollo de Aplicaciones con Android

Santiago Faci

Centro Afuera

Junio 2019

- 1 ¿Qué es Android?
- 2 Estructura de una aplicación Android
 - Estructura del Sistema Operativo
 - Ciclo de vida de una aplicación
 - Estructura de una Aplicación
 - Demo v1
- 3 Datos y Mapas
 - Datos abiertos
 - Mapbox
 - Demo v2
- 4 Acceso a Bases de Datos
 - SQLite
 - Demo v3

¿Qué es un framework?

framework

Un framework es un conjunto de programas, bibliotecas, lenguajes y otras herramientas, que sirve de base para el desarrollo de software.

En función del framework utilizado, se creará un tipo u otro de aplicación

Android Framework / Sistema Operativo

Android SDK

Android SDK es el kit de desarrollo que permite crear aplicaciones para este Sistema Operativo

Android SDK funciona como un framework que, utilizando lenguaje Java, permite desarrollar aplicaciones apoyándose en multitud de librerías y herramientas ya creadas con una estructura definida.

Android Framework / Sistema Operativo

Android SDK

Android SDK es el kit de desarrollo que permite crear aplicaciones para este Sistema Operativo

Android SDK funciona como un framework que, utilizando lenguaje Java, permite desarrollar aplicaciones apoyándose en multitud de librerías y herramientas ya creadas con una estructura definida.

Android Sistema Operativo

Android es además un Sistema Operativo (basado en Linux) para dispositivos móviles (tablets, smartphones, smartwatches, ...) que viene instalado en numerosos terminales

Estructura de Android

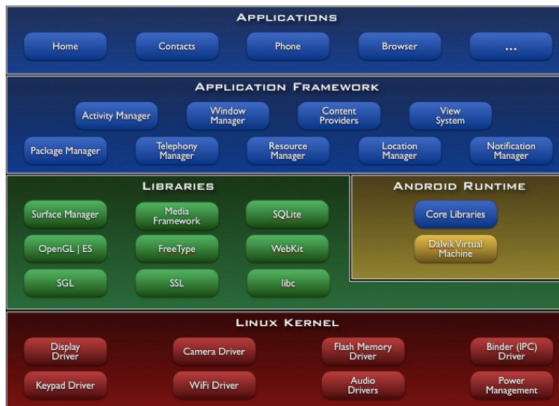


Figura: Framework Android

Ciclo de vida de una aplicación Android

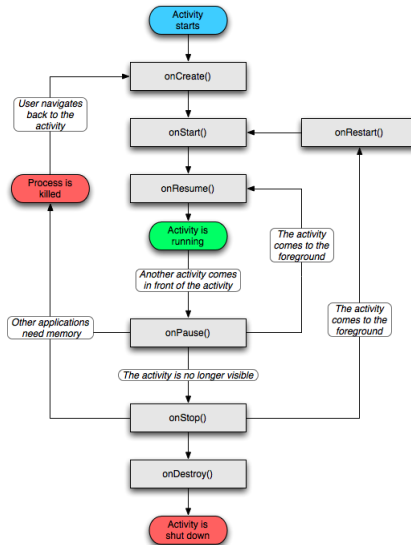


Figura: Ciclo de vida

Estructura del proyecto

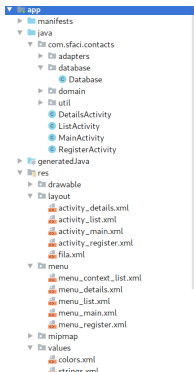


Figura: Proyecto Android

Estructura del proyecto

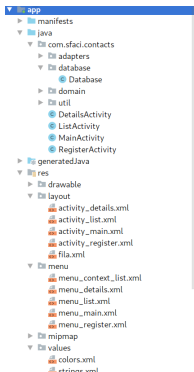


Figura: Proyecto Android

- **java** Código de aplicación
- **res** Recursos del proyecto
- **drawable** Iconos de la aplicación
- **layout** Diseño de las pantallas
- **menu** Diseño de los diferentes menús
- **values** Recursos de texto
- **manifests** Almacena el manifiesto del proyecto con la configuración general del mismo

Recursos de aplicación

Los recursos son todos aquellos ficheros que no forman parte del código de la aplicación. Se localizan dentro de la carpeta *res* en la estructura del proyecto y se distribuyen a su vez en carpetas

Recursos de aplicación

Los recursos son todos aquellos ficheros que no forman parte del código de la aplicación. Se localizan dentro de la carpeta *res* en la estructura del proyecto y se distribuyen a su vez en carpetas

- *drawable* Repetida con algunos de estos sufijos (*xhdpi*, *hdpi*, *ldpi*, *mdpi*), almacena los recursos gráficos del proyecto en sus diferentes resoluciones (según sufijo)

Recursos de aplicación

Los recursos son todos aquellos ficheros que no forman parte del código de la aplicación. Se localizan dentro de la carpeta *res* en la estructura del proyecto y se distribuyen a su vez en carpetas

- *drawable* Repetida con algunos de estos sufijos (*xhdpi*, *hdpi*, *ldpi*, *mdpi*), almacena los recursos gráficos del proyecto en sus diferentes resoluciones (según sufijo)
- *layout* Diseños de las diferentes pantallas de la aplicación

Recursos de aplicación

Los recursos son todos aquellos ficheros que no forman parte del código de la aplicación. Se localizan dentro de la carpeta *res* en la estructura del proyecto y se distribuyen a su vez en carpetas

- *drawable* Repetida con algunos de estos sufijos (*xhdpi*, *hdpi*, *ldpi*, *mdpi*), almacena los recursos gráficos del proyecto en sus diferentes resoluciones (según sufijo)
- *layout* Diseños de las diferentes pantallas de la aplicación
- *values* Cualquier otro recurso que no sea código (textos de la aplicación, opciones, . . .)

Recursos de aplicación

Los recursos son todos aquellos ficheros que no forman parte del código de la aplicación. Se localizan dentro de la carpeta *res* en la estructura del proyecto y se distribuyen a su vez en carpetas

- *drawable* Repetida con algunos de estos sufijos (*xhdpi*, *hdpi*, *ldpi*, *mdpi*), almacena los recursos gráficos del proyecto en sus diferentes resoluciones (según sufijo)
- *layout* Diseños de las diferentes pantallas de la aplicación
- *values* Cualquier otro recurso que no sea código (textos de la aplicación, opciones, . . .)
- *menu* Diseño de los menús de la aplicación

Código de aplicación

Código desarrollado

El código desarrollado se empaqueta como una aplicación *Java*.

Código generado

Todos los recursos del proyecto generan, automáticamente, una clase *R* para utilizar éstos desde el código de la aplicación

Fichero de manifiesto I

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
  package="org.sfaci.helloworld"  
  android:versionCode="1"  
  android:versionName="1.0">
```

Define el paquete base de la aplicación y el número y nombre de versión de la misma

Fichero de manifiesto II

```
<uses-sdk android:minSdkVersion="18"/>
```

Indica la versión mínima de Android (*API Level*) soportada por la aplicación (requisitos mínimos para funcionar)

- Android 2.3.3 → *API Level 10*
- Android 3.0 → *API Level 11*
- ...
- Android 4.3 → *API Level 18*
- Android 4.4 → *API Level 19*

Fichero de manifiesto III

```
<application android:label="@string/app_name" android:icon="@drawable/  
    ic_launcher">
```

Define el nombre de la aplicación y su icono

Fichero de manifiesto IV

```
<activity android:name="MyActivity"
    android:label="@string/app_name">
    <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
    </intent-filter>
</activity>
```

- Define cada una de las *Activity* que componen la aplicación
- Solamente una se debe definir como *Activity* principal (la del ejemplo en este caso)

Empaquetado de la aplicación

El empaquetado de la aplicación lo realiza la propia aplicación generando un fichero .apk que se puede instalar directamente en cualquier móvil Android.

- Cada vez que ejecutamos la aplicación sobre el emulador o nuestro dispositivo para depurar, se genera automáticamente una versión empaquetada de la misma que sólo puede ser utilizada para esos fines.
- Se debe firmar con nuestro fichero de claves si queremos distribuir la aplicación y subirla a Google Play.

Demo Agenda v1

Aplicación Android que sirve como agenda de contactos:

- Registro de nuevos contactos
- Listado de contactos
- Comunicación entre Activities
- Adaptador personalizado para listar los contactos
- Llamada a algunas funciones del dispositivo móvil
- Información no persistente (no hay base de datos)

Datos abiertos

Es posible conectarse a Internet para obtener información de fuentes abiertas (datos abiertos)

- La información normalmente vendrá en formato JSON
- Algunas ciudades tienen un catálogo completo de datos abiertos sobre eventos o información relativa a la misma
- Normalmente estará actualizada y fácilmente disponible. Sólo hay que parsearla correctamente.

Mapas

Se pueden utilizar muchas de las librerías existentes, además de Google Maps. En este caso utilizaremos la librería Mapbox, que podemos usar gratuitamente mientras nuestra aplicación no supere los 25.000 usuarios activos mensuales o las 50.000 cargas mensuales de mapas.

- Disponible en <http://www.mapbox.com>
- Tiene toda una API disponible para realizar todo tipo de acciones sobre el mapa y también para extraer información

Demo Agenda v2

Aplicación Android que sirve como agenda de contactos e información de eventos

- Registro de nuevos contactos
- Listado de contactos (aún sin base de datos)
- Uso de datos abiertos para mostrar información en el dispositivo
- Visualización de información sobre un mapa Mapbox

¿Qué es SQLite?

- Motor de Base de Datos de tamaño muy reducido (unos pocos MB)
- Ideal para Bases de Datos *pequeñas* (aprox. 1 GB)
- Perfecto para pequeños dispositivos
- No necesita instalación (un único fichero .jar en el caso de Java)
- Viene *de serie* con Android
- Fácil de usar (sólo hay que saber *SQL* y manejarse con *JDBC*)
- Muchas aplicaciones lo usan (*Whatsapp*, por ejemplo)

Demo Agenda v3

Aplicación Android que sirve como agenda de contactos almacenando la información en Base de Datos

- Registro de nuevos contactos en base de datos
- Listado de contactos de la base de datos
- Sin necesidad de escribir SQL para las operaciones más habituales
- Información persistente (utilizando SQLite)