

Programando Android - Estructura de aplicación

José Antonio Vacas

#android2014
03/12/2014



ElCacharreo.com



Programación en Android



José Antonio Vacas Martínez

blog
javacasm@elcacharreo.com
twitter
linkedin



Programación en Android: Recursos

ANDROID developer.android.com

Muy bueno http://www.sgoliver.net/blog/?page_id=3011

Avanzado <http://www.limecreativelabs.com/curso-gratuito-de-desarrollo-para-android/>

Avanzado(En) <http://www.vogella.com/android.html>

MiriadaX http://miriadax.net/es/web/android_programacion

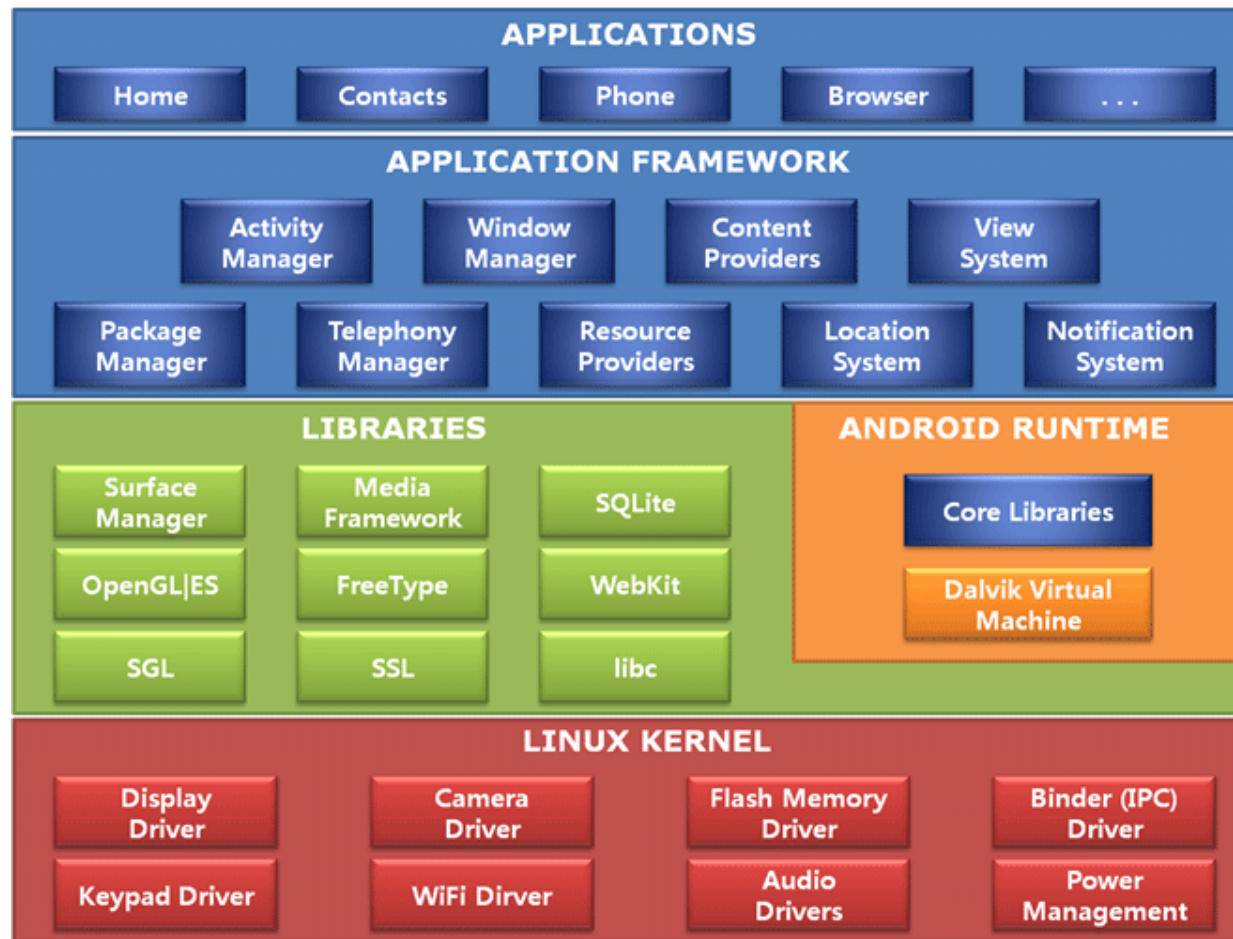


Programación en Android: Objetivos

- Conocer la plataforma Android
- Entorno de programación
- Crear proyecto de aplicación básica
- Crear "pantallas"
- Almacenamiento
- Publicación de aplicaciones



Programación en Android: Arquitectura



[Detalle](#)

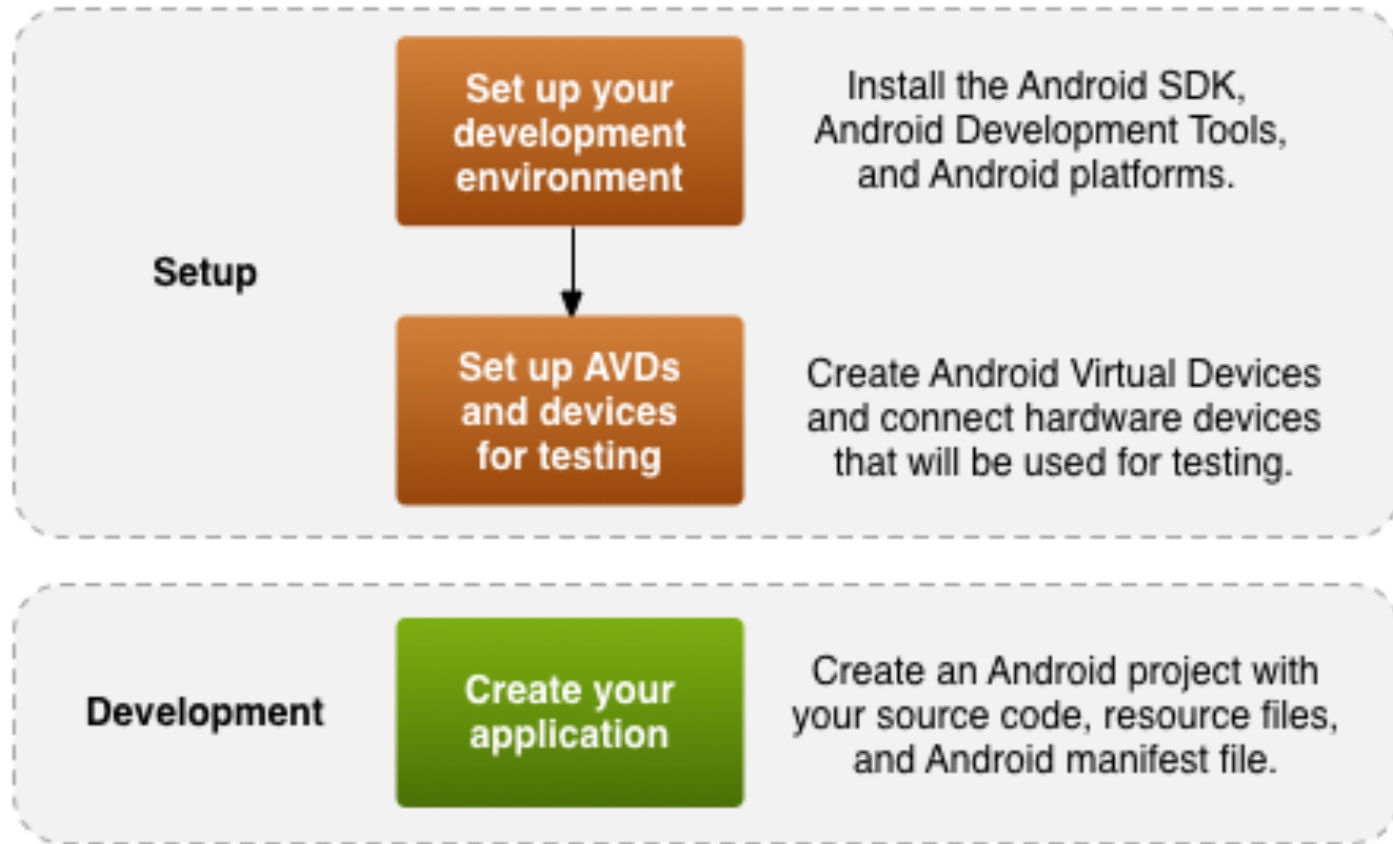


Programación en Android: Arquitectura

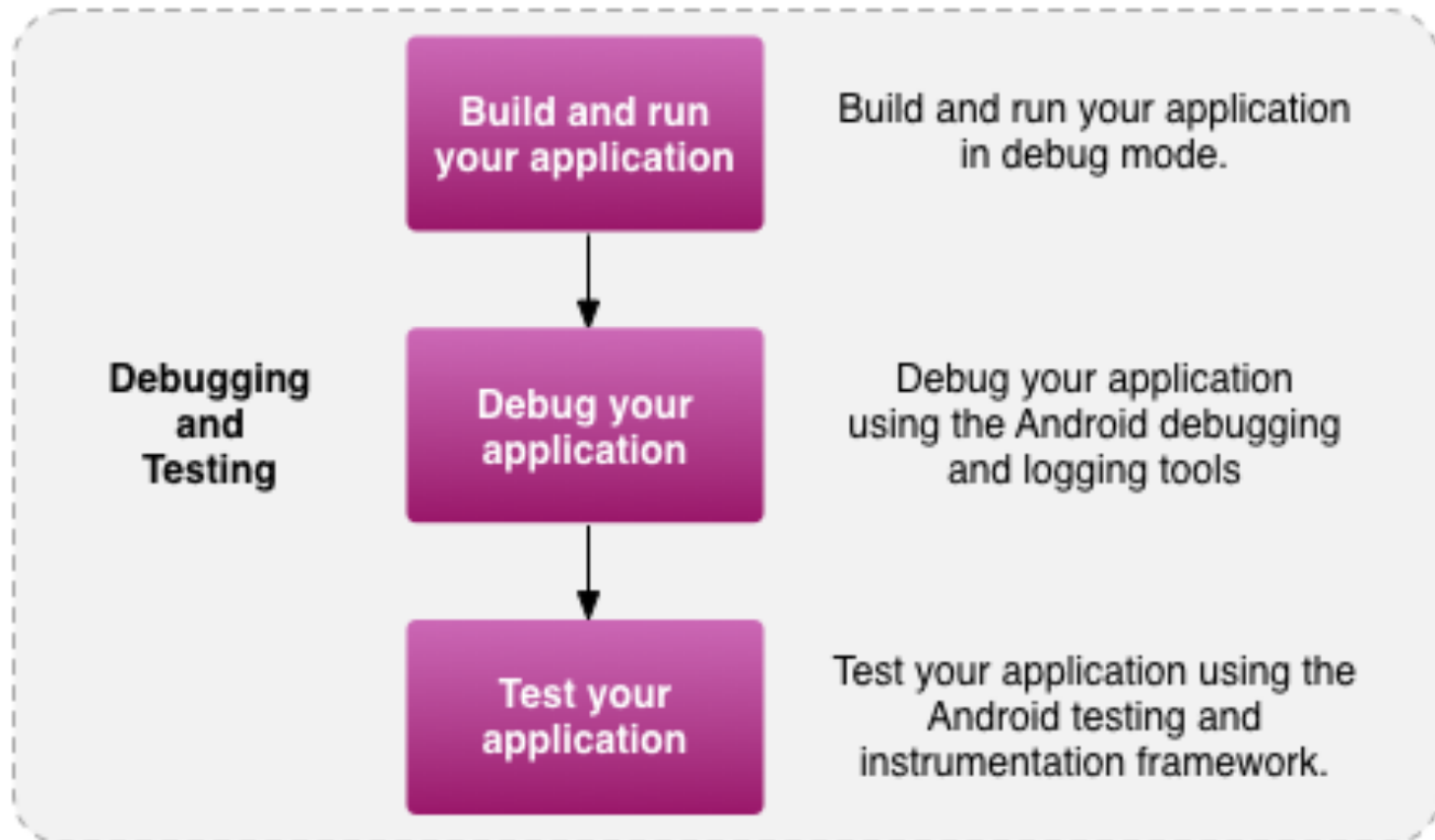
- Cada aplicación se ejecuta en su propia máquina virtual Dalvik
- Dalvik basada en el kernel de linux, poca memoria, bytecode dex
- Hasta 2.2 el bytecode era interpretado, ahora JIT (Just In Time)



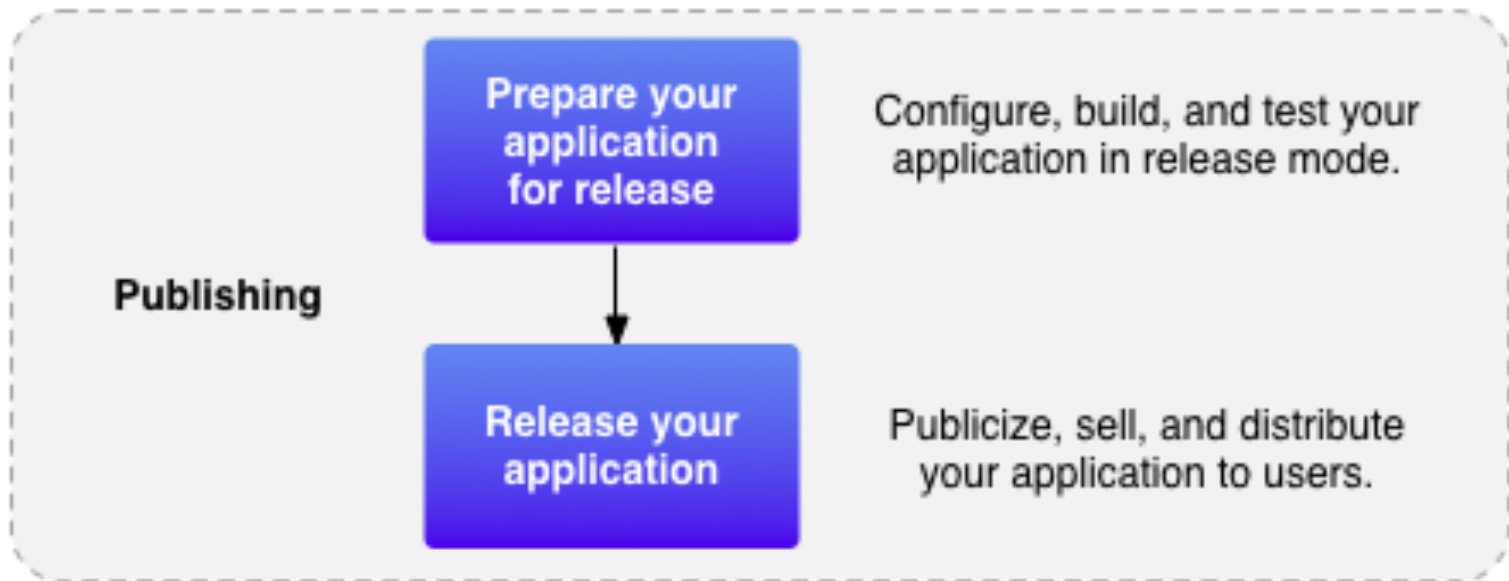
Programación en Android: Desarrollo



Programación en Android: Desarrollo



Programación en Android: Desarrollo



Programación en Android: Arquitectura

Framework de aplicación

- **Views** (vistas): se utilizan para construir una aplicación, incluyendo lists (listas), grids (rejillas), text box (cajas de texto), buttons (botones), e incluso un navegador web embebido
- **Content Providers** (proveedores de contenido) que permiten a las aplicaciones acceder a datos de otras aplicaciones (como los contactos), o compartir sus propios datos
- **Resource Manager** (administrador de recursos), facilitar el acceso a los recursos no son de código tales como cadenas localizadas, gráficos y archivos de diseño
- **Notification Manager** (Administrador de notificaciones) que permite a todas las aplicaciones mostrar alertas personalizadas en la barra de estado
- **Activity Manager** (gestor de actividad) que gestiona el ciclo de vida de las aplicaciones y proporciona una navegación común



Programación en Android: Arquitectura

- **Activities:** manejan la interfaz de usuario y la pantalla del smartphone. Se van añadiendo a una cola LIFO
- **Services:** manejan los procesamiento en segundo plano.
- **Broadcast receivers:** manejan la comunicación entre sus aplicaciones.
- **Content providers:** manejan los datos y todo lo relacionado con la gestión de datos.
- **Intent:** contiene un mensaje que se envía entre los diferentes módulos.



Programación en Android: Arquitectura

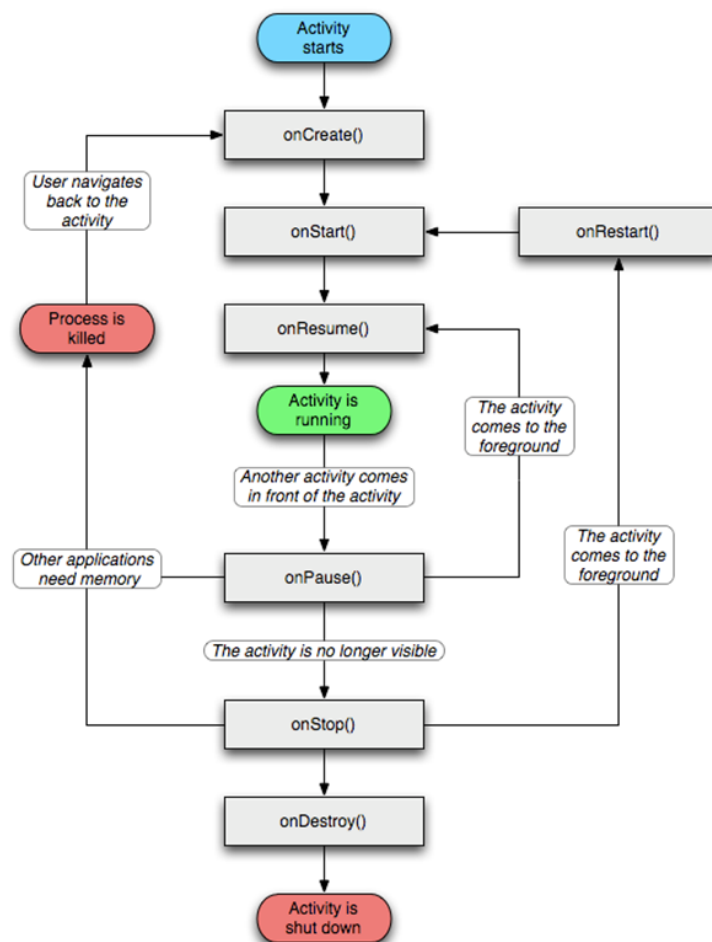
Fichero Manifest.xml

El fichero manifest sirve para muchos más propósitos que simplemente definir los componentes de nuestra aplicación. La siguiente lista resume las partes relevantes de un fichero manifest en el contexto del desarrollo de juegos:

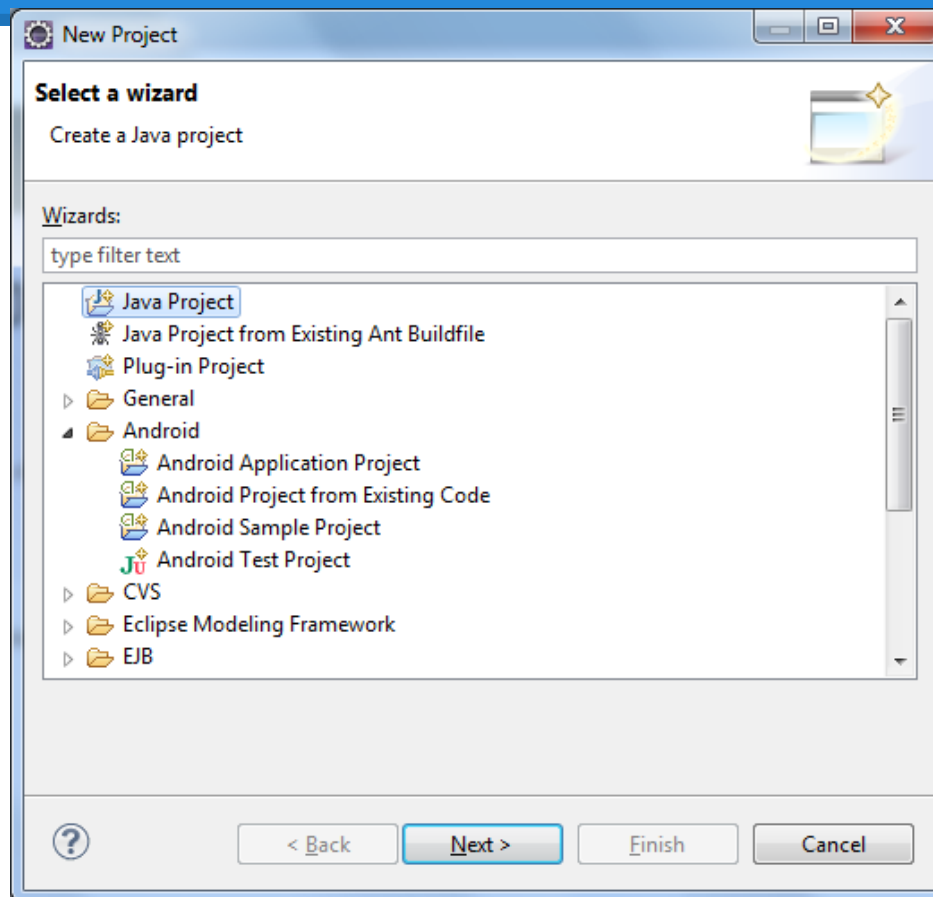
- La versión de nuestra aplicación tal y como se muestra y utiliza en el Market de Android
- La versión de Android en la que nuestra aplicación puede funcionar
- Perfiles de hardware que nuestra aplicación requiere (ie: multitáctil, resoluciones de pantalla específicas o soporte para OpenGL ES 2.0)
- Permisos para usar componentes específicos, tales como escritura en la tarjeta SD o acceso a la pila de red.



Programación en Android: Estructura de aplicación



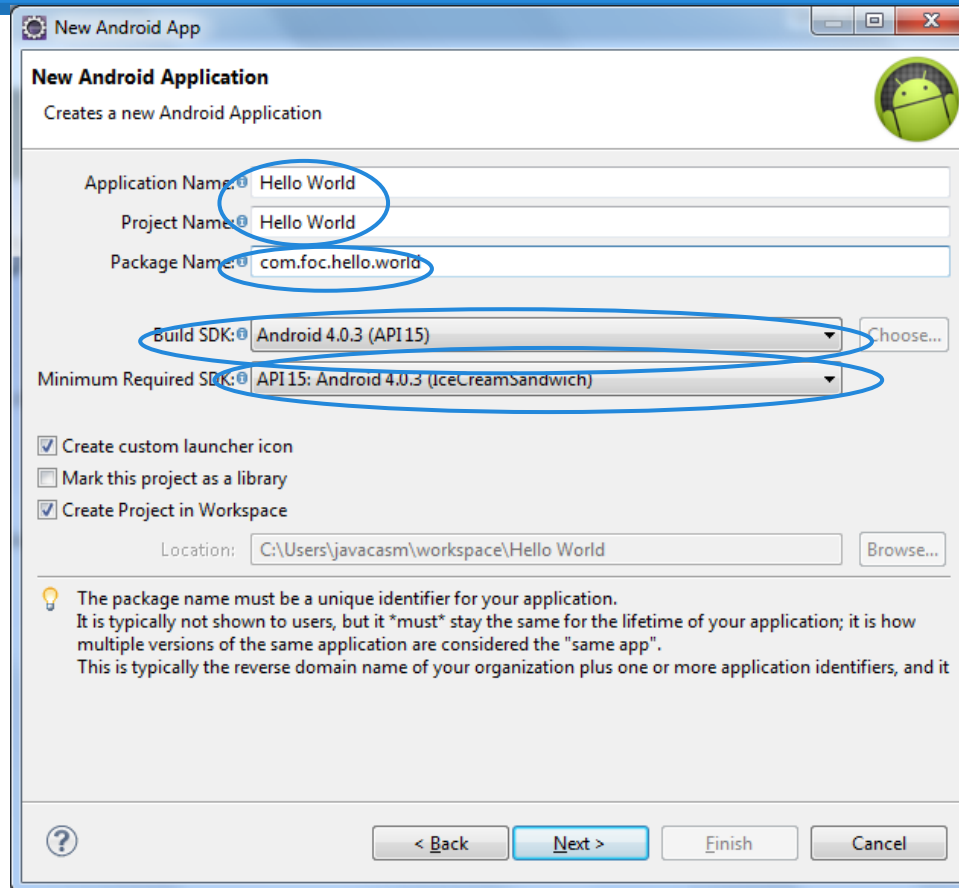
Programación en Android: Primeros pasos



Pulsamos sobre New Project y ahí seleccionamos Android Application Project



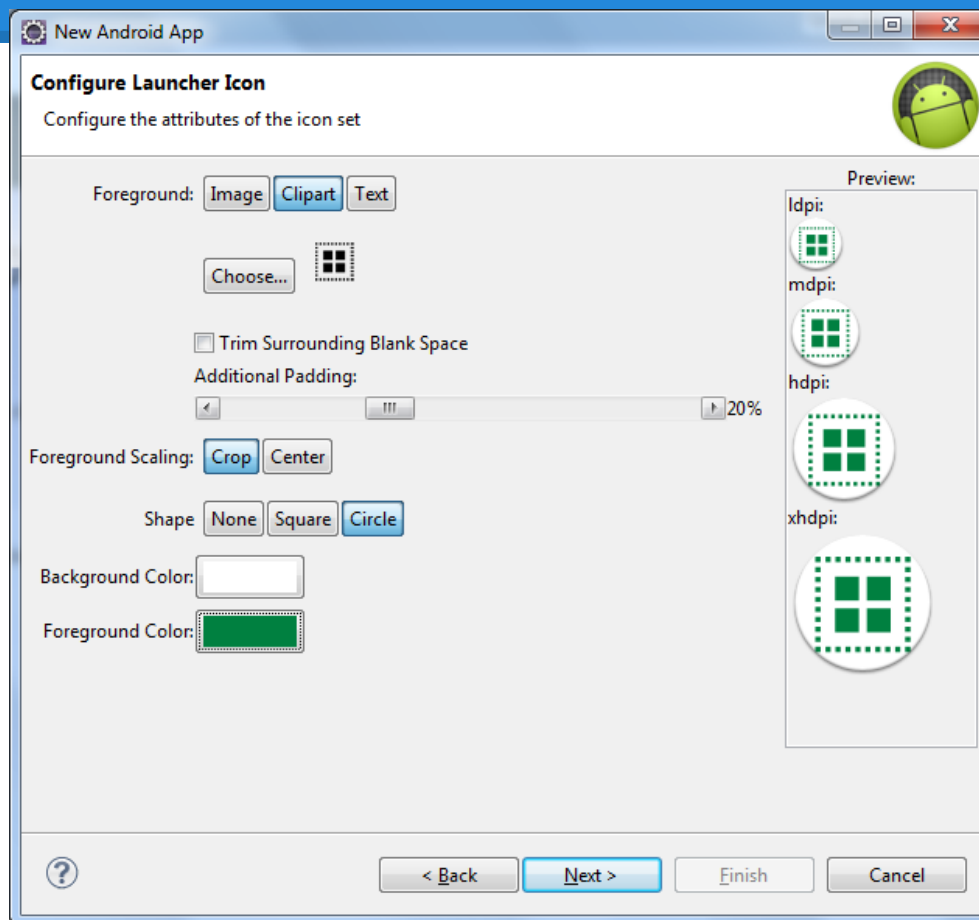
Programación en Android: Primeros pasos



Ahora asignamos las propiedades fundamentales del proyecto



Programación en Android: Primeros pasos

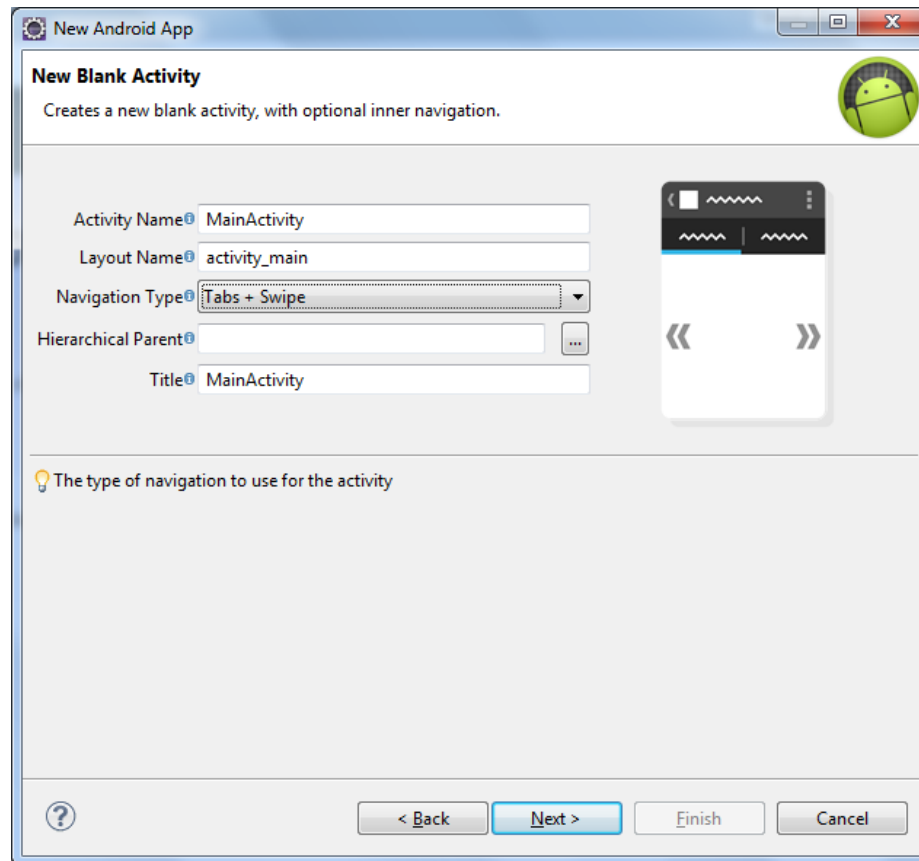


Podemos configurar el icono de nuestra aplicación, a partir de una imagen

ElCacharreo.com

#android2014

Programación en Android: Primeros pasos



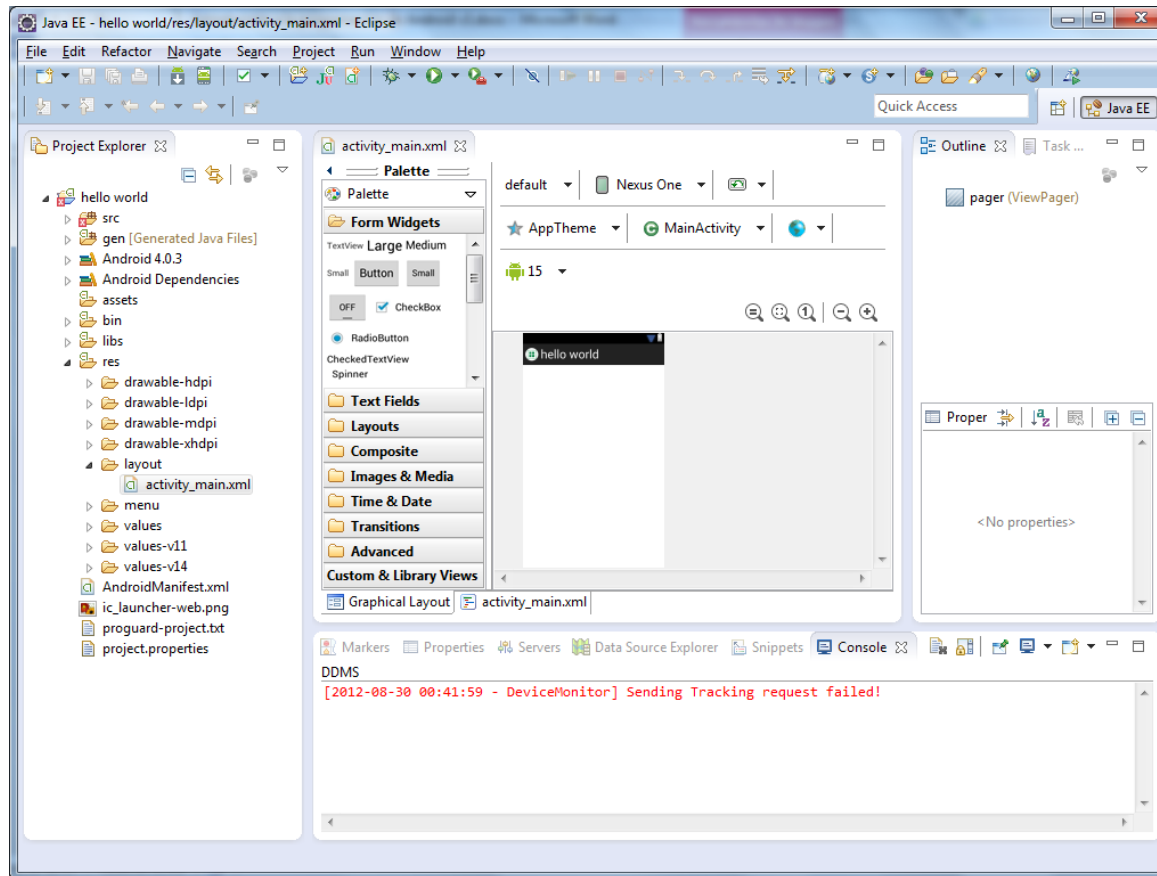
Ahora seleccionaremos el nombre de esta activity y la forma de la navegación de la aplicación

ElCacharreo.com

#android2014



Programación en Android: Primeros pasos

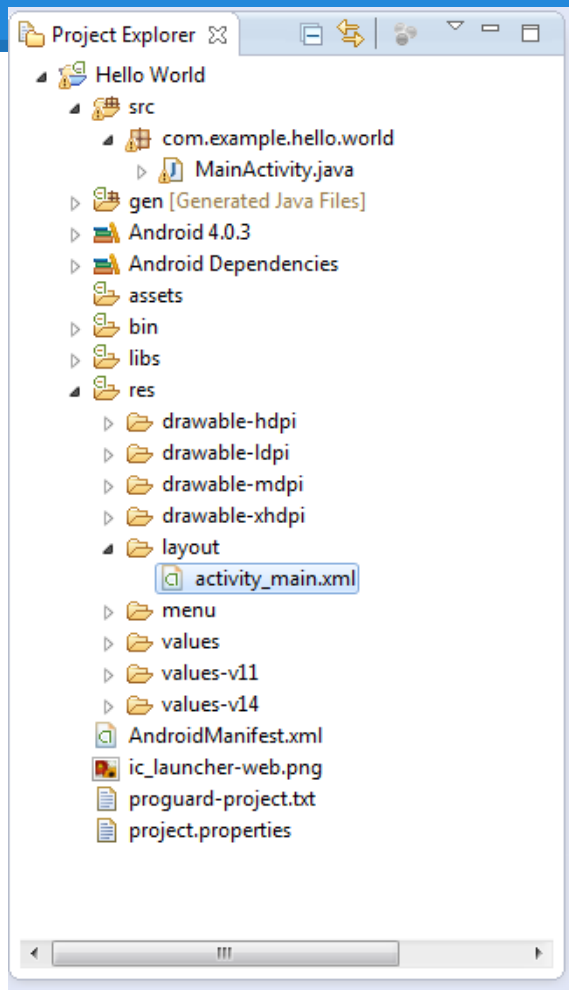


Si todo ha ido correctamente, tendremos cargado nuestro nuevo proyecto



Programación en Android

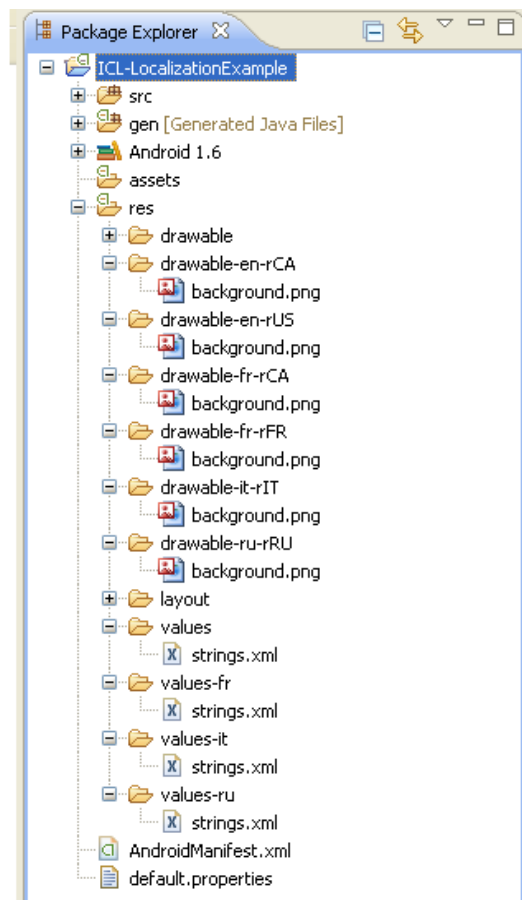
Estructura de proyecto



- **AndroidManifest.xml** describe su aplicación. Define de qué actividades y servicios está compuesto, qué versiones mínima y objetivo de Android se suponen que van a ejecutarse y qué permisos necesitan
- **project.properties** contiene varias configuraciones para construir el sistema. No debemos tocar aquí ya que el plug-in de ADT lo modifica cuando es necesario
- **src/** contiene todos los ficheros fuente Java. Note que el paquete tiene el mismo nombre que el nombre del proyecto Android
- **gen/** contiene los ficheros fuente Java generados por el sistema de construcción de Android. Estos se generan automáticamente en algunos casos. El más importante es el que se puede observar en la imagen, el fichero **R.java**, y la clase **R**. Esta clase **R** contendrá en todo momento una serie de constantes con los ID de todos los recursos de la aplicación incluidos en la carpeta **/res/**, de forma que podamos acceder fácilmente a estos recursos desde nuestro código a través de este dato. Así, por ejemplo, la constante **R.drawable.icon** contendrá el ID de la imagen "icon.png" contenida en la carpeta **/res/drawable/**
- **assets/** es donde se almacenan los ficheros que nuestra aplicación necesite (i.e. ficheros de configuración, ficheros de audio, etc.). Estos ficheros se empaquetarán con la aplicación Android
- **res/** contiene recursos que la aplicación necesite como iconos, cadenas de texto para internacionalización, apariencia del UI definidos vía XML. También se empaquetarán con la aplicación Android
- **Android 4.0.3** indica que se está generando contra una versión de Android 4.0.3. Esto es normalmente una dependencia en la forma de un JAR estándar que contiene las clases de la API Android 4.0.3.
- El Explorador de Paquetes contiene otro directorio llamado **bin/** que aloja el código compilado que se necesita para el dispositivo o el emulador. Al igual que la carpeta **gen/** no nos preocupamos mucho de ellos.



Programación en Android: Recursos - Traducciones



Del mismo modo en el caso de que tengamos imágenes distintas dependiendo del idioma duplicaremos la carpeta para cada idioma con los diferentes contenidos.

En el ejemplo:

Italian	drawable-it-rIT/background.png
French	drawable-fr-rFR/background.png
French (Canada)	drawable-fr-rCA/background.png
English (Canada)	drawable-en-rCA/background.png
Russian	drawable-ru-rRU/background.png
US English	drawable-en-rUS/background.png
Default (Earth image)	drawable/background.png



Programación en Android: Recursos

ANDROID developer.android.com

Muy bueno http://www.sgoliver.net/blog/?page_id=3011

Avanzado <http://www.limecreativelabs.com/curso-gratuito-de-desarrollo-para-android/>

Avanzado(En) <http://www.vogella.com/android.html>

MiriadaX http://miriadax.net/es/web/android_programacion

