

Tecnologías y Desarrollo en Dispositivos Móviles

Apartado 3: **Arquitectura del Sistema Operativo Android.** **Publicación de apps.**

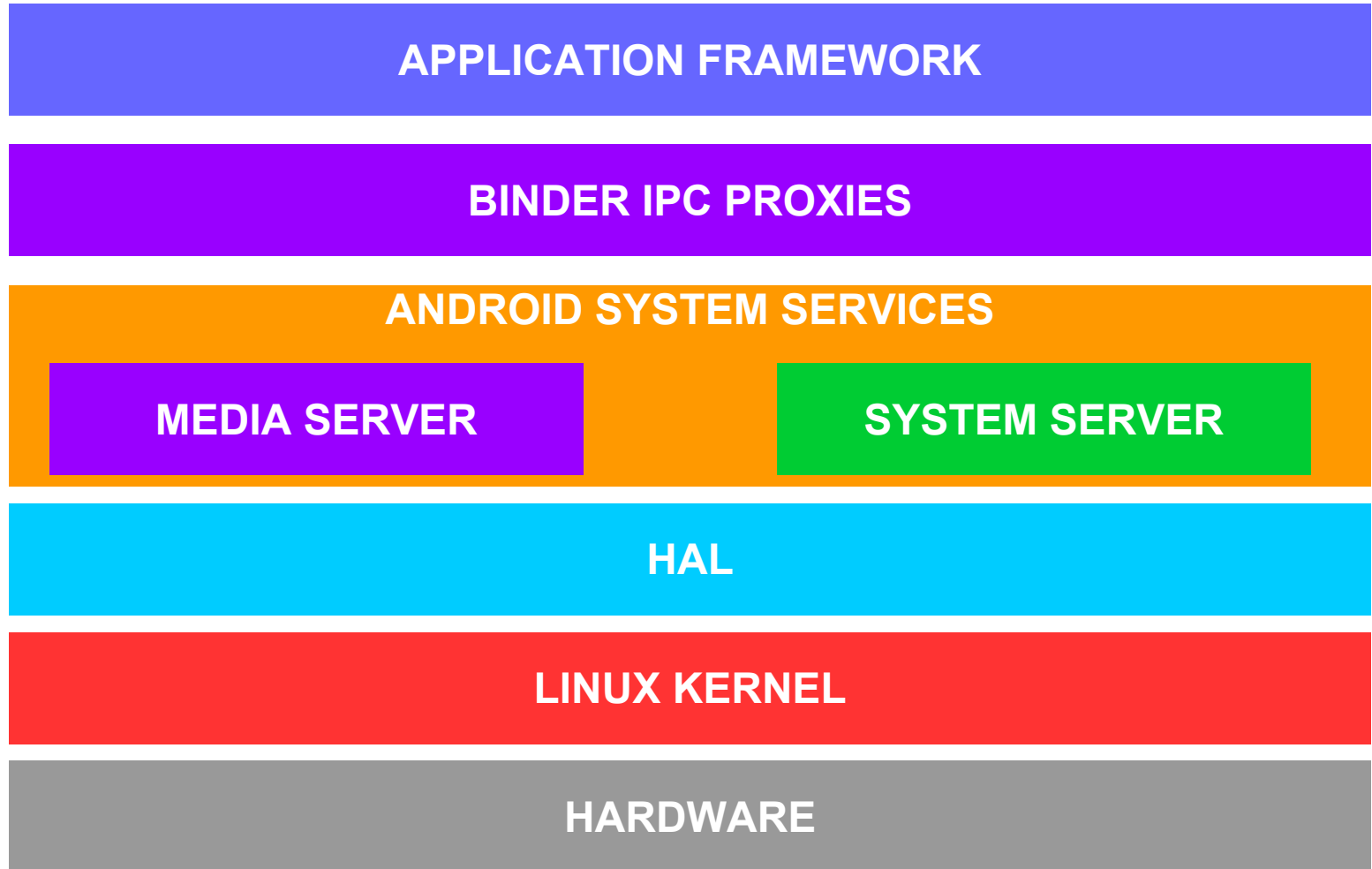
Autores:

Víctor M. Rivas Santos
(Juan Ruiz de Miras)

Software en dispositivos Android

- Sistema operativo: versión modificada del kernel de linux
- Middleware: conecta el SO a bajo nivel con las aplicaciones
- ART: jre para ejecutar las aplicaciones
- Aplicaciones “básicas”: navegador, contactos, teléfono, etc.
- Resto de Apps

Arquitectura del Sistema



Application Framework y Binder IPC Proxies

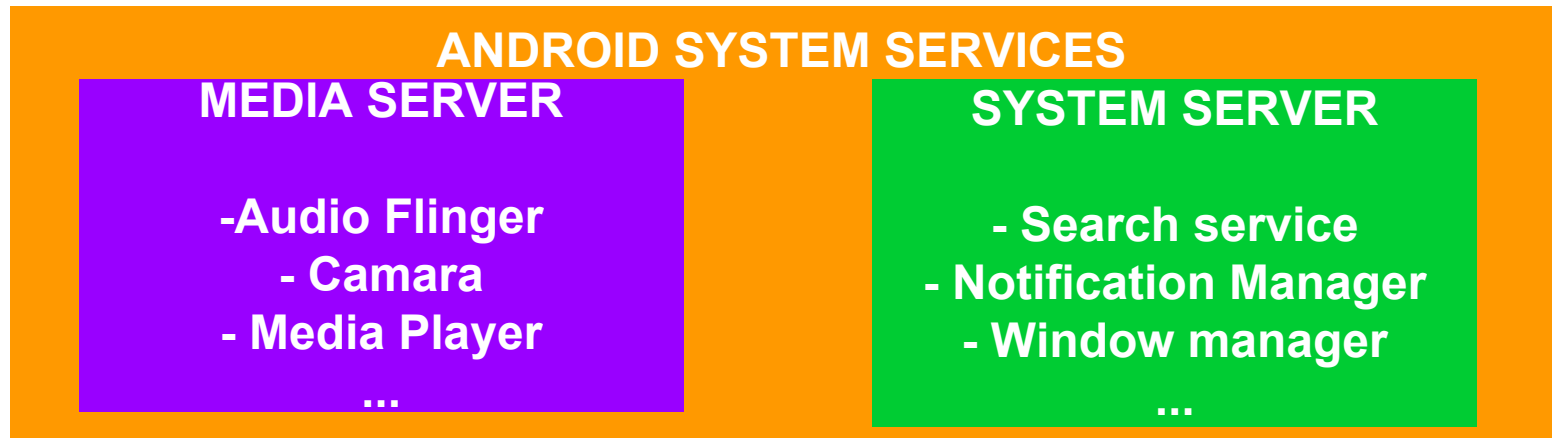
APPLICATION FRAMEWORK

- Espacio en el que se ejecutan las apps que crearemos

BINDER IPC PROXIES

- *Binder Inter-Process Communication*: permiten a las apps llamar a los servicios que provee el *Android System Services*. Es transparente para nosotros como desarrolladores

Android System Services y HAL

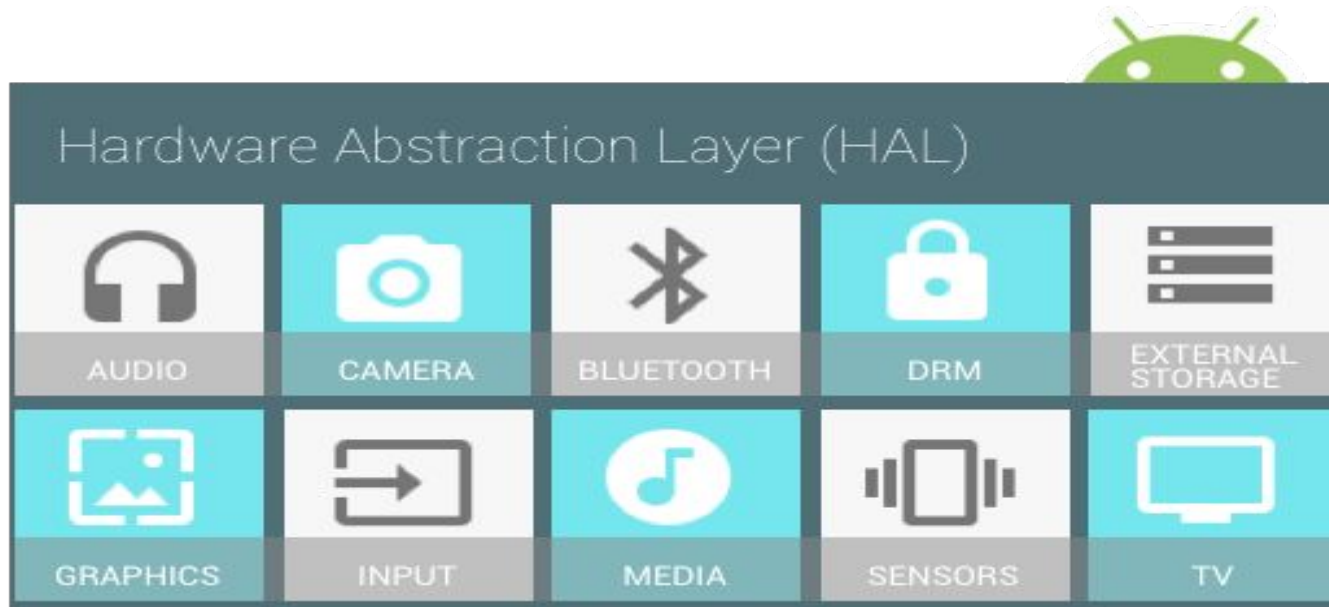


- Servicios que provee Android a las apps mediante APIs



- *Hardware abstraction layer* (HAL): define estándares para los proveedores de hardware, de modo que el SO no tenga que preocuparse de los detalles de cada uno

Componentes HAL



Fuente: <https://source.android.com/devices/architecture/hal>

- Android 8.0
 - Binderized HALs
 - Passthrough HALs

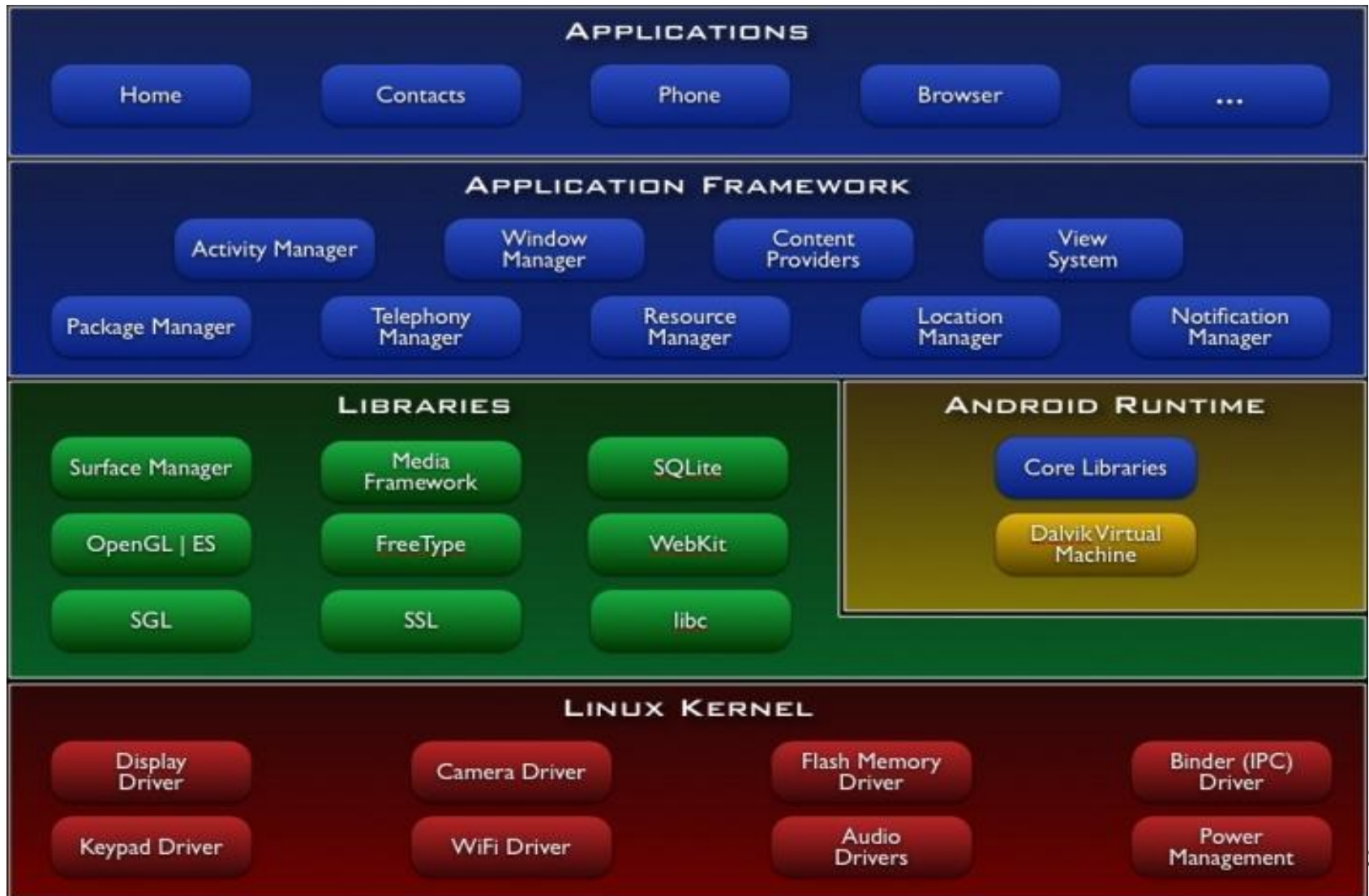
Fuente: <https://source.android.com/devices/architecture/hal-types/>

Linux Kernel

LINUX KERNEL

- Kernel del SO y drivers para los dispositivos:
 - cámara,
 - audio (ALSA, OSS; etc),
 - Pantalla,
 - etc

“Arquitectura antigua Android”



Detalles del *System Server*

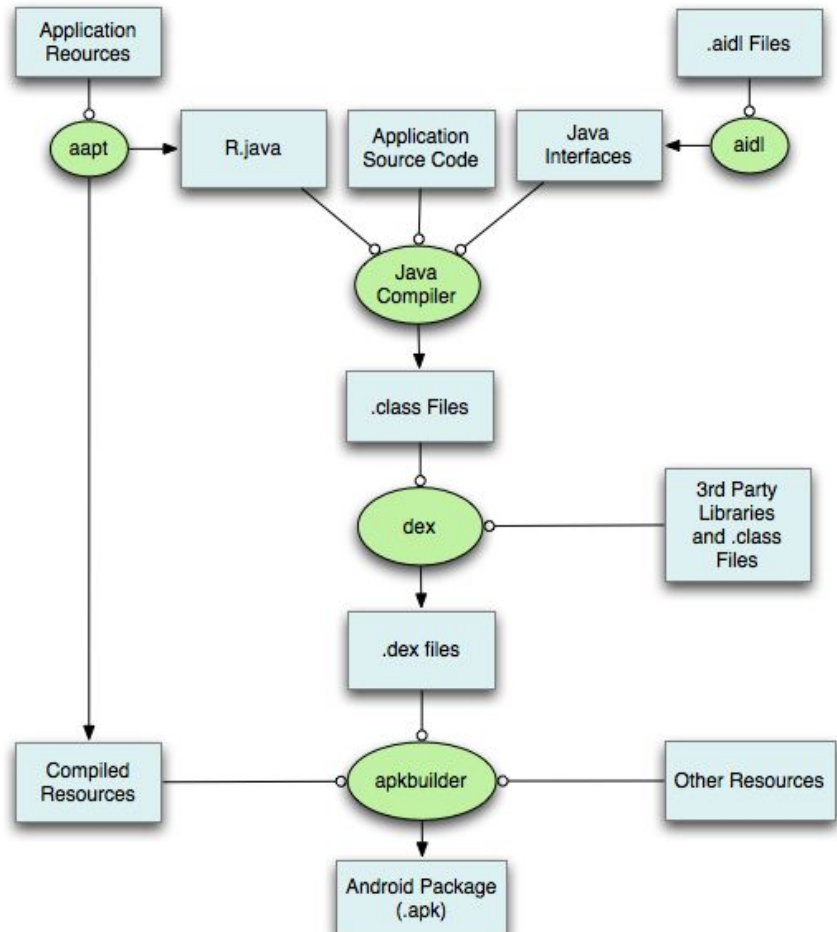
- . *View system*
 - *Lists, grids, buttons...*
- . *Resource manager*
 - *Layouts, images, strings...*
- . *Activity manager*
 - Ciclo de vida, *backstack*
- . *Notification manager*
 - Gestión de alertas personalizadas
- . *Content provider*
 - Contactos, favoritos ...

```
Uri uri = Uri.withAppendedPath(  
    CommonDataKinds.Contactables.CONTENT_FILTER_URI, query);
```

<https://developer.android.com/reference/android/provider/ContactsContract.CommonDataKinds.Contactables.html>

Empaquetado .apk

- 1) *aapt* compila los ficheros de recursos, AndroidManifest.xml, ficheros XML de actividades, y crea el fichero R.java. Si hay ficheros .aidl se compilan con *aidl*
- 2) Se compila el código Java generando los .class
- 3) *dex* convierte los .class a Dalvik bytecode
- 4) *apkbuilder* empaqueta los ficheros .dex y resto de recursos (como imágenes o sonidos) en un fichero .apk
- 5) Finalmente, se puede firmar como *debug* o *release* antes de ser instalado.



<http://stackoverflow.com/questions/24570534/how-and-what-converts-java-bytecode-to-android-dex-file#24570735>

El jre ART

- Introducido en KitKat, sustituyó a Dalvik
- Mediante *dex2oat*: compila ficheros dex empaquetados en el .apk

Ventajas y desventajas de ART

MTV: Justificar el cambio de Dalvik a ART

- Realiza un pequeño documento en el que:
 - Recojas las ventajas y desventajas de ART frente a Dalvik.
 - Indiques tu opinión acerca de si fue justificado o no el cambio realizado a partir de la versión Kit Kat.
 - No olvides aportar referencias a la bibliografía o enlaces que hayas utilizado y subir el documento a la plataforma en formato PDF.

Descomprimir fichero apk

MTV: Comprobar qué se instala en el dispositivo exactamente

- Descarga el fichero apk que hemos preparado
- Descomprímelo (es posible que tengas que cambiar su extensión a .zip)
- Observa la estructura de directorios y las extensiones de ficheros que contiene
 - ¿Has localizado los ficheros compilados (.dex)?
 - ¿puedes abrir alguno de los ficheros XML?
- Prueba ahora a descomprimirlo con la herramienta *apktool* (solo necesitas descargar el *.jar* desde <https://bitbucket.org/iBotPeaches/apktool/downloads>)
 - ¿puedes ahora abrir los ficheros XML?
- Sube a la plataforma una captura de pantalla del contenido del fichero apk

Publicación de apps

- La principal vía de publicación de aplicaciones Android es a través de Google Play (antes Android Market)
- Google Play permite:
 - buscar y descargar aplicaciones, libros, música y películas
 - puntuar, comentar, desinstalar y actualizar las aplicaciones instaladas en el dispositivo

Generación y firma del apk

- . Se puede generar directamente con Android Studio
 - Build > Build APK
 - Se almacena *Directorio_del_Proyecto/App/build/outputs/apk*
 - Se crea en modo debug, Android Studio firma por defecto la aplicación para poder ejecutarla en los emuladores o en los dispositivos conectados al IDE.
 - Este apk no se puede publicar ni distribuir.
- . Para publicar o actualizar una app hay que firmar el .apk
 - Generate signed APK
 - Es necesario un certificado (.jks), se puede crear directamente desde Android Studio
 - el archivo .apk obtenido puede instalarse directamente en el dispositivo y durante toda la vida de la aplicación (actualizaciones) hay que firmar con el mismo certificado

Proceso de publicación en Google Play

- . Abrir una cuenta de desarrollador:
 - Dirección de Gmail
 - Número de teléfono
 - 25\$
- . Contrato de desarrollador:
 - Precio lo establece el desarrollador
 - Google cobra una comisión de transacción (30% del precio de venta) y descuenta impuestos locales
 - Dinero obtenido = precio de venta - comisión – impuestos
 - Soporte de la aplicación lo da el desarrollador

<https://play.google.com/apps/publish>

Prohibiciones expresas

- Apps en Google Play no pueden tener:
 - contenido sexual
 - contenido violento o xenófobo
 - suplantación de identidad o información engañosa
 - contenido con protección intelectual
 - divulgación de información personal
 - venta ilegal
 - software malicioso: virus, spyware, ...
 - acceso al dispositivo, salvo los permisos solicitados
 - uso elevado de la red (sobrecostes al usuario)
 - prácticas de spam

Proceso de publicación

- . Se sube el fichero APK
- . Datos necesarios de la aplicación:
 - Dos capturas de pantalla
 - Icono a 512 x 512 píxeles
 - Idiomas soportados por la aplicación
 - Título y descripción de la aplicación en cada idioma soportado
 - Tipo de aplicación (juego o aplicación)
 - Categoría (compras, deportes, educación, fotografía, ...)
 - Tipo de público al que va dirigida la app
 - Países en los que la app estará disponible

Seguimiento y actualización en Google Play

- . Estadísticas para el desarrollador de cada aplicación publicada:
 - Puntuación media
 - Detalle de puntuaciones y comentarios de usuarios
 - Errores reportados por los usuarios
 - Número de instalaciones totales
 - Número de instalaciones activas (app sigue en el dispositivo)
 - Instalaciones por idioma, por país, por dispositivo, por versión de Android, ...
- . Cuando se sube una nueva versión de una aplicación existente hay que indicar las novedades en cada idioma soportado

Antes de publicar una app

- . Probarla en:
 - un dispositivo con la mínima versión requerida
 - un dispositivo con plataforma de destino desarrollada
 - un dispositivo con la última versión Android
 - dispositivos reales, no emuladores
 - diferentes tamaños y resoluciones de pantalla
 - en el modo *landscape*, si está soportado por la app:
 - . Se puede elegir que sólo se ejecute en un modo. En manifiesto, en etiqueta del activity:

```
android:screenOrientation="{portrait | landscape}"
```
 - en condiciones de poca o nula conectividad a la red

Antes de publicar una app (II)

- . Eliminar o poner a false la propiedad *android:debuggable* de la etiqueta *<application>* del manifiesto
- . establecer en el manifiesto (etiqueta *<manifest>*) las propiedades:
 - *android:versionCode*: número entero que representa la versión de la aplicación, versiones posteriores deben tener valores mayores. Se utiliza para establecer si la versión instalada en un dispositivo es la última disponible o no. Ej: 1, 2, 3, 4, ...
 - *android:versionName*: cadena que representa la versión de la aplicación, esta información se muestra al usuario, aunque sólo se utiliza a título informativo. Ej: “1.0”, “1.1”, “2.3”, ...

Otras formas de publicar apps

- Publicando el .apk en una página web
- Distribuyendo el .apk mediante e-mail o similar
- Android por defecto impide instalación de aplicaciones que no procedan de Google Play:
 - Avisar a usuarios que desactiven este bloqueo: Ajustes -> Seguridad -> Fuentes Desconocidas
 - Avisar de que vuelvan a activarlo una vez instalado
- Otros mercados donde publicar aplicaciones Android:
 - [Opera Mobile Store](#)
 - [Amazon Appstore](#)

Amazon Appstore

- Permite subir apps para Android, Amazon Fire, Mac y PC, y Fire TV
- Necesitamos abrir una cuenta de desarrollador.
- Antes de subir la app:
 - Comprobar el checklist previo a la publicación para estar seguro de que cumple con los requerimientos de Amazon y las guías de estilo
 - Hacer un test previo (mediante la herramienta que ofrece Amazon) a la app

<https://developer.amazon.com/app-submission>

Test de APK en Amazon

Test Results

testmaria.apk | Feb-15-2017 5:42 PM 

[View Dashboard](#)

Compatibility Test



Compatibility test successfully completed. We did not find any of the common issues that can prevent an app from being published to the Amazon Appstore

[Submit to Amazon Appstore](#)

Amazon Appstore (II)

- . Proceso para subir la app:
 - Paso 1: Acceder a la plataforma con la cuenta de desarrollador y “crear la app”
 - Paso 2: Introducir información genérica de la app
 - Paso 3: Editar la información relativa al precio
 - Paso 4: Introducir la descripción detallada de la app
 - Paso 5: Aportar capturas de imagen y vídeos de funcionamiento
 - Paso 6: Asignar categorías al contenido de la app, para que pueda clasificarse mejor
 - Paso 7: Subir el .apk
 - Paso 8: Asociar la app a un perfil de seguridad (solo si se utilizan los servicios de GameCircle o Amazon Device Messaging)
 - Paso 9: Enviar usando el botón “Submit”

<https://developer.amazon.com/app-submission>

Test de app en Amazon

MTV: Comprobar el servicio de test de apps gratuito de Amazon

- . Crea una aplicación tan sencilla como desees
- . Genera el correspondiente .apk, sin firmar, y comprueba si puedes pasarle el test de compatibilidad de Amazon.
- . En caso afirmativo, no cierres la página de Amazon dado que tarda bastante en generar la compatibilidad por dispositivos
- . Cuando termine, o cuando te canses de esperar, sube un documento PDF que contenga el informe completo del Test (puedes usar la opción de imprimir del navegador) si ha dado tiempo, o al menos el mensaje de que el mensaje de compatibilidad ha terminado satisfactoriamente