

Tema 2. Entorno de desarrollo



Prof. Manuel Castillo

Programación de Dispositivos Móviles

Escuela Profesional de Ciencias de la Computación

Facultad de Ciencias

Universidad Nacional de Ingeniería

Objetivos



- Conocer los entornos de desarrollo de Android.
- Saber cómo manejar proyectos.
- Saber cómo descargar los componentes principales del SDK.
- Saber cómo actualizar el entorno de desarrollo.
- Saber cómo crear una máquina virtual de Android para poder ejecutar las aplicaciones.
- Saber cómo arrancar una maquina virtual de Android.

Índice de contenido



- Introducción.
- Android Studio.
- Herramientas comunes.
- Estructura de ficheros



1. Introducción

1. Introducción



- Existen dos herramientas de desarrollo principales
 - Eclipse+ADT: herramienta en desuso.
 - Android Studio: herramienta oficial.
- Herramientas comunes
 - SDK Manager: Manejo de paquetes de componentes Android.
 - AVD Manager: Manejo de máquinas virtuales.



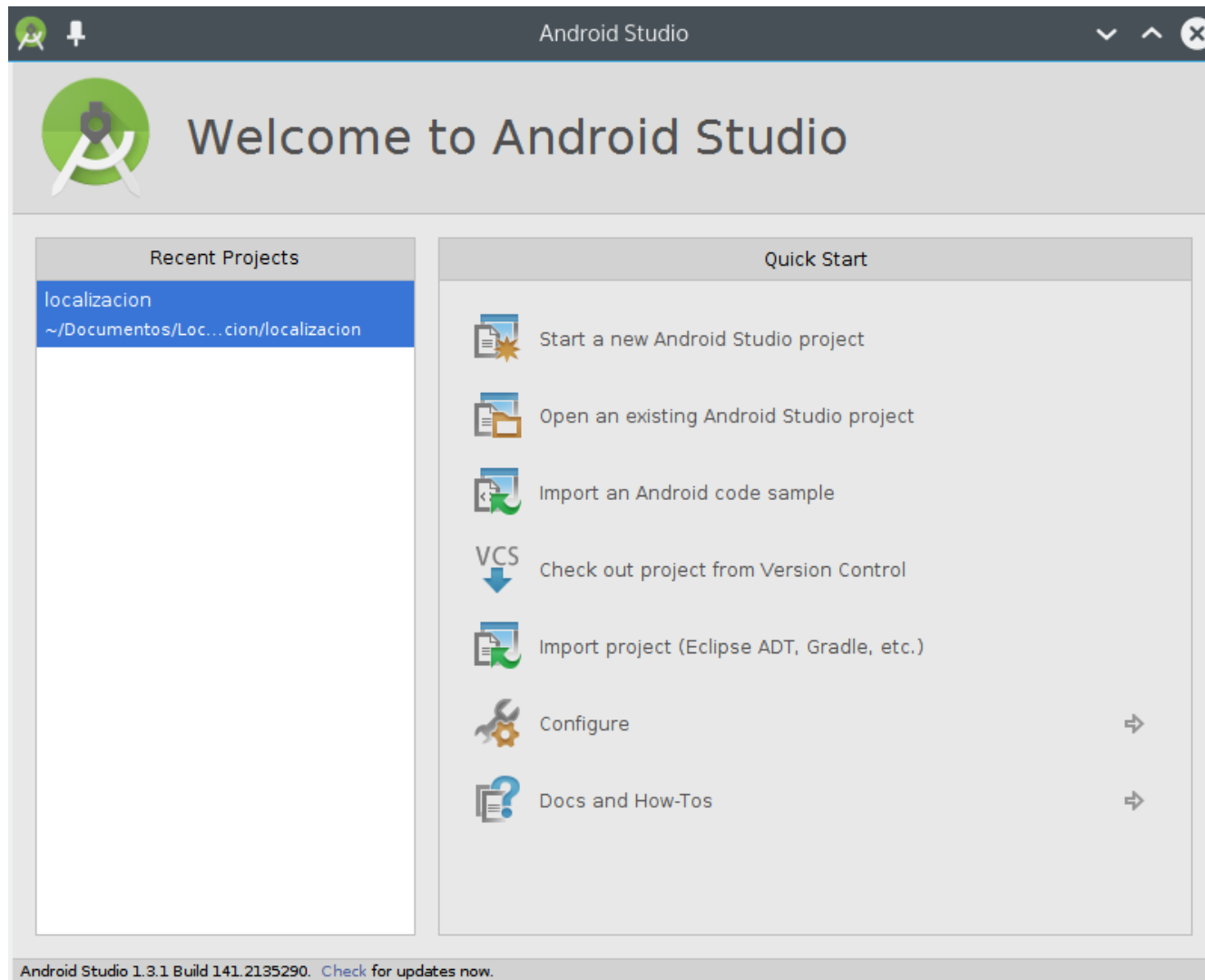
2. Android Studio

2.1. Características



- SO: Ubuntu, MacOS X, Windows
- Java Development Kit – JDK
- Android Studio
 - Entorno basado en IntelliJ IDEA: editor robusto y maduro.
 - Plataforma de desarrollo oficial.
 - Compilación basada en Gradle, Eclipse basado en ANT.
 - Compilación estable y con más control.
 - Android SDK.

2.2. Pantalla de bienvenida



2.3. Información



Create New Project

New Project

Android Studio

Configure your new project

Application name:

Company Domain:

Package name: [Edit](#)

Project location: ...

[Previous](#) [Next](#) [Cancel](#) [Finish](#)

Create New Project

Target Android Devices

Select the form factors your app will run on

Different platforms may require separate SDKs

☒ Phone and Tablet
Minimum SDK: [Help me choose](#)
Lower API levels target more devices, but have fewer features available. By targeting API 15 and later, your app will run on approximately **94,0%** of the devices that are active on the Google Play Store.

☐ Wear
Minimum SDK:

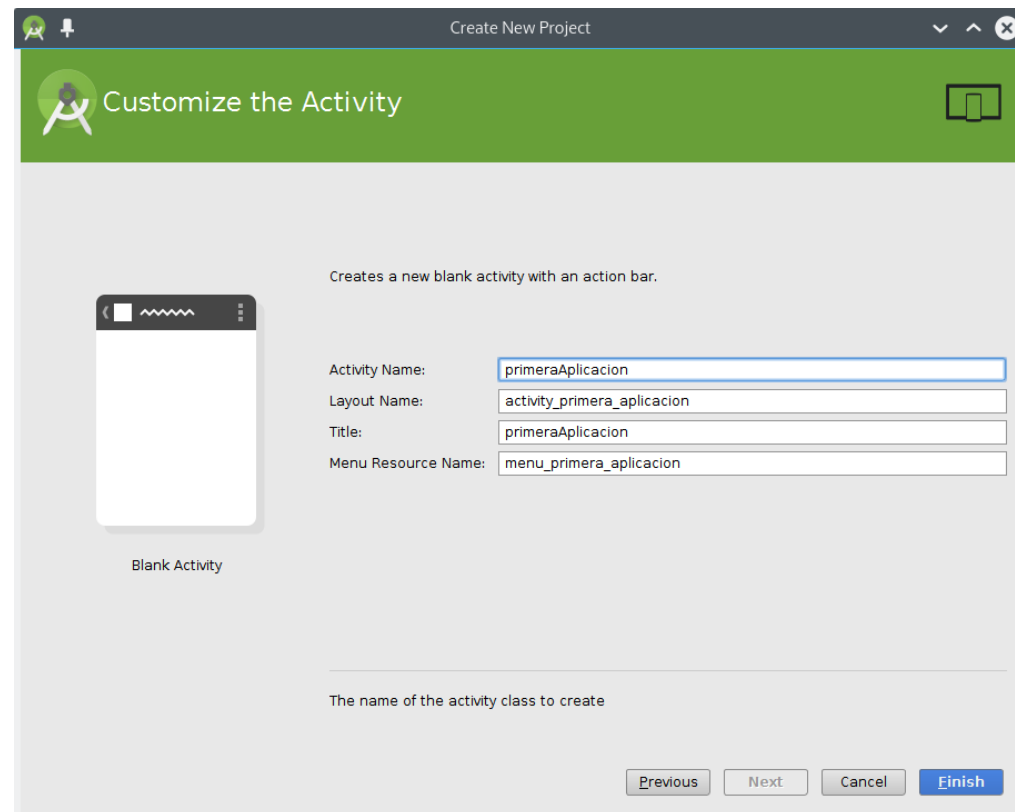
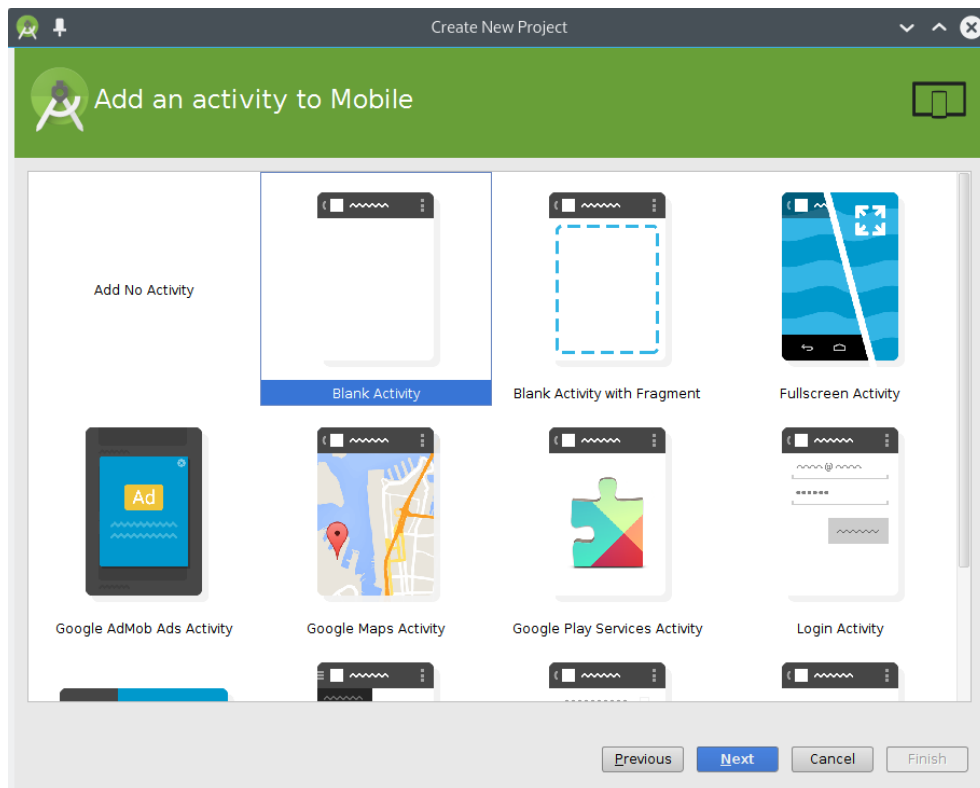
☐ TV
Minimum SDK:

☐ Android Auto

☐ Glass
Minimum SDK:

[Previous](#) [Next](#) [Cancel](#) [Finish](#)

2.4. Añadir actividad



2.5. Entorno inicial



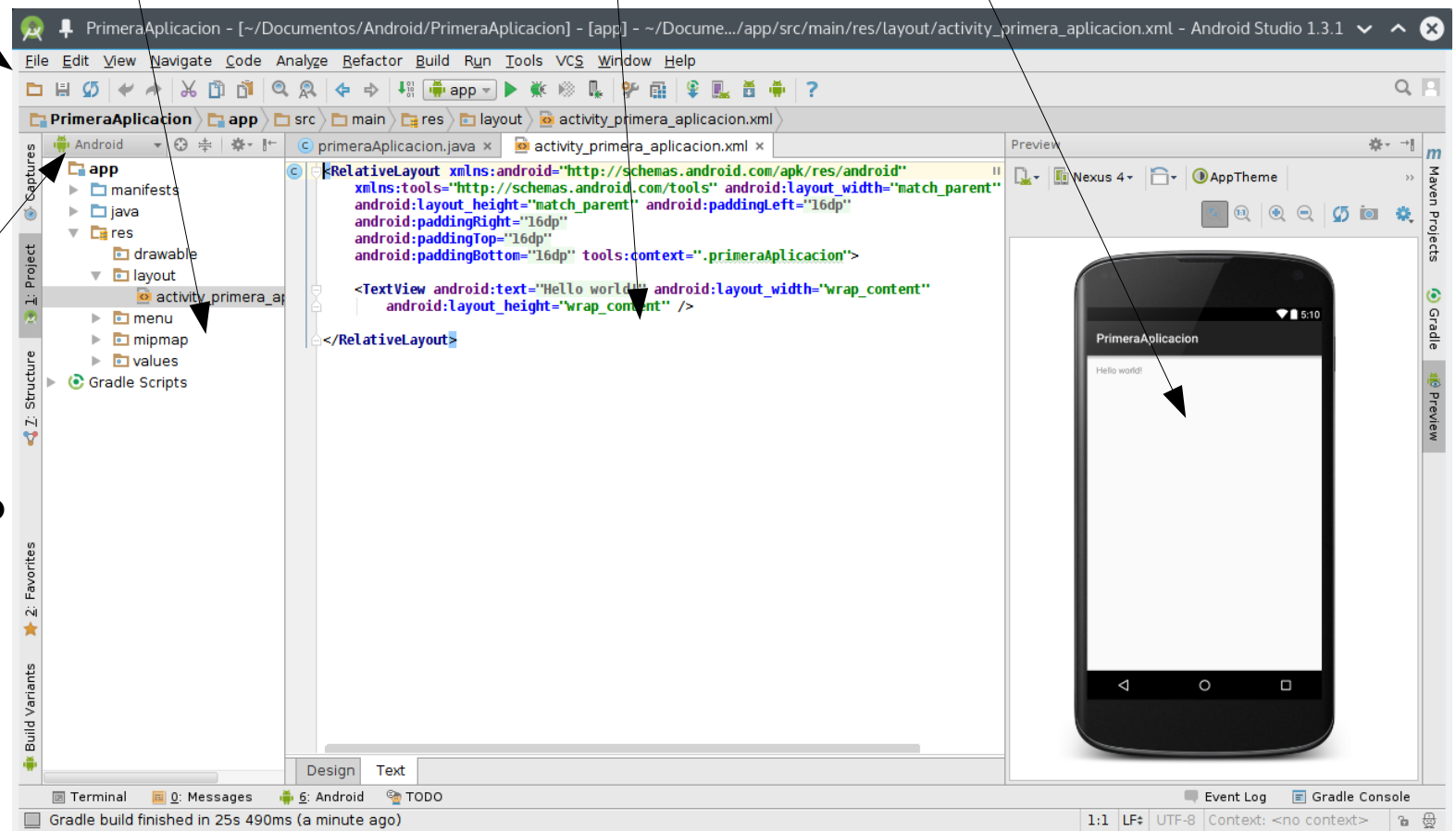
Explorador de Archivos/paquetes

Edición de ficheros

Visualización previa

Iconos de Acceso directo

Vistas de nuestro proyecto

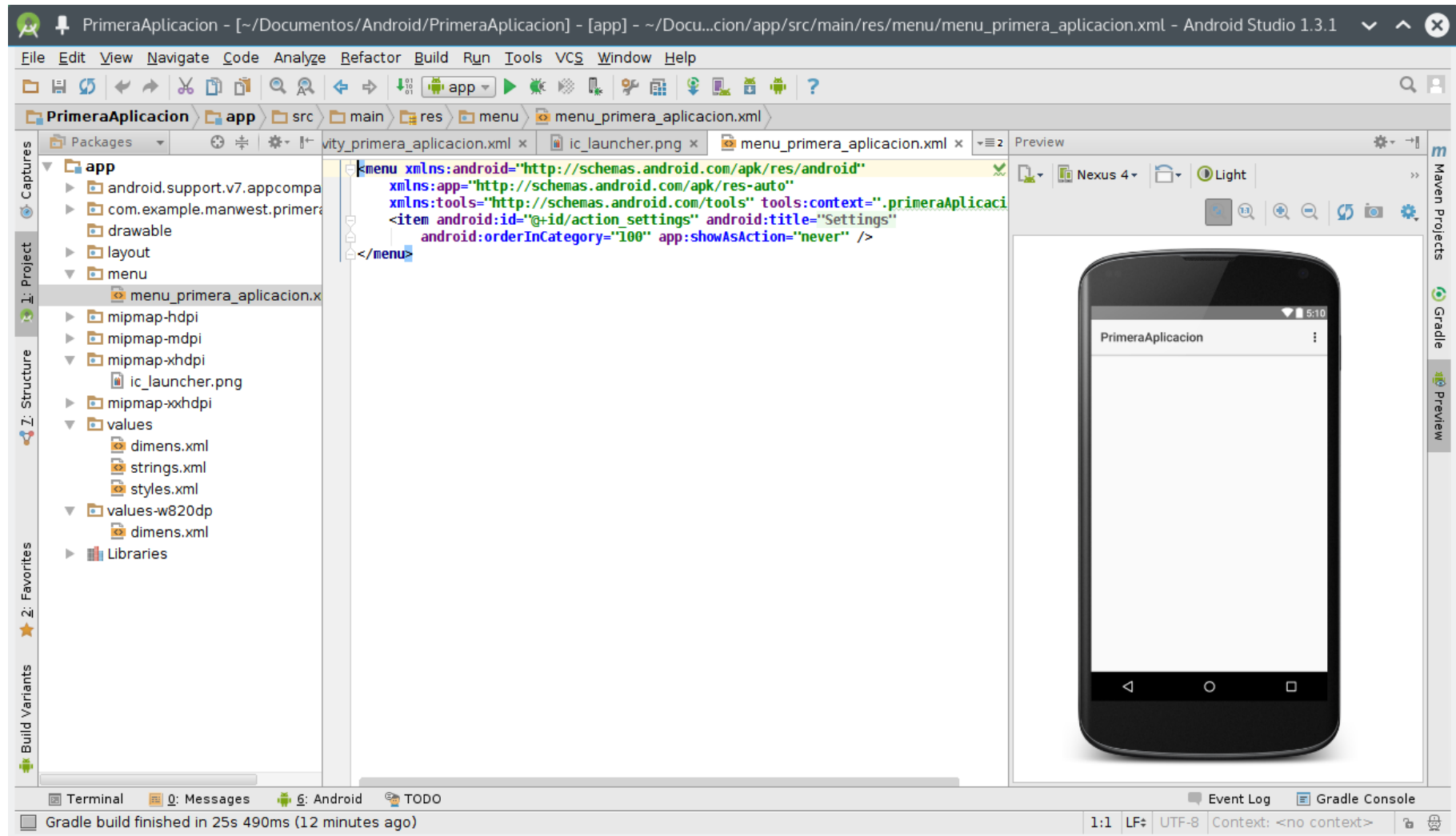


2.6. Sobre Android Studio



- Tiene una visualización distinta de los proyectos.
- Es una ventana por proyecto.
- La vista de proyecto (Comando+1) es lo más similar al “Project explorer” en Eclipse.
- Lo similar al contenido del proyecto Eclipse es la carpeta app, que tiene una estructura similar.
- Dentro de la vista de proyecto podemos abrir una vista de resumen con un desplegable y poniendo Packages en vez de Project
- La perspectiva DDMS (Debug) se abre automáticamente cuando se va a ejecutar una aplicación.

2.7. Vista Package



2.7. Vista Package



- Vista de simplificación.
- Drawable: imágenes
- Layout: diseño de las pantallas.
- Menu: menús de la aplicación.
- Mindmap: icono de nuestra aplicación
- Values: gestión de literales y estilos.
- Libraries: biblioteca.



3. Herramientas Comunes

3.1. Introducción

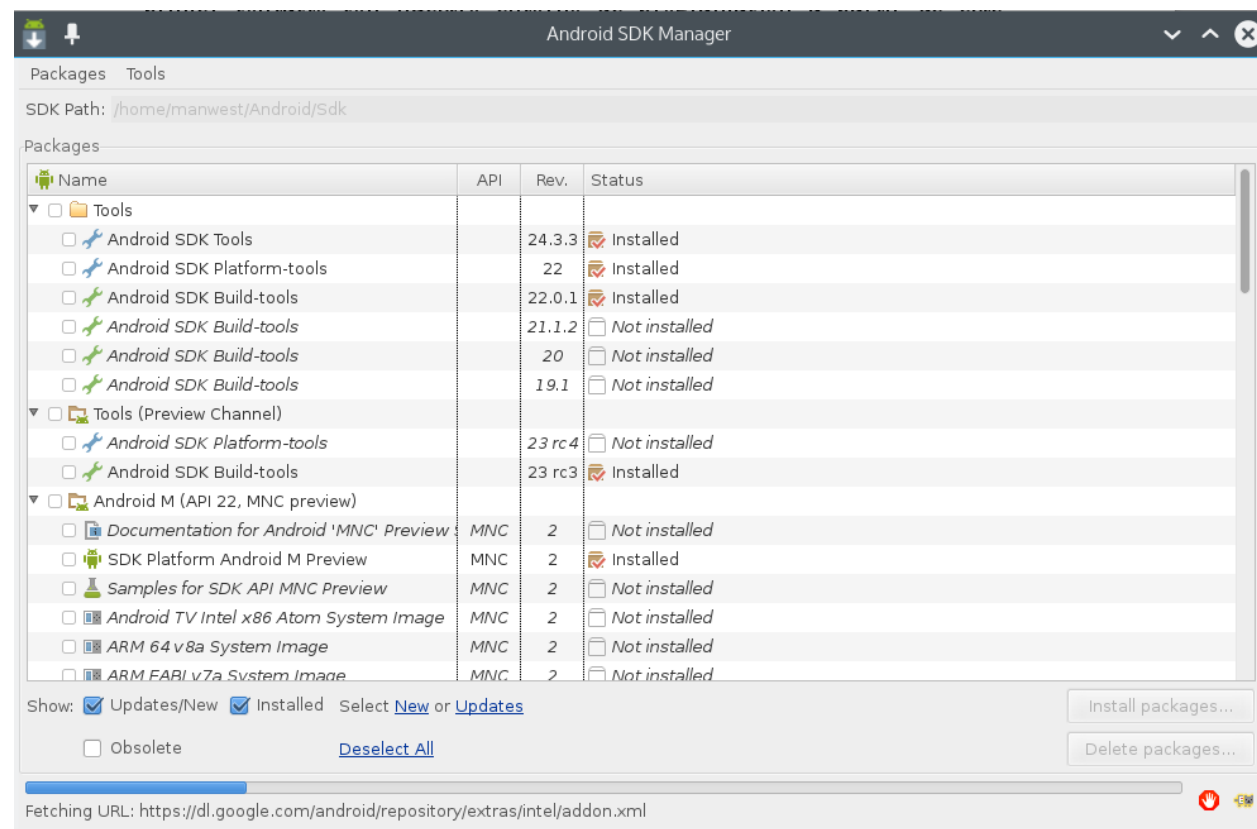


- El SDK manager está disponible desde el entorno de desarrollo:
 - Es el botón con la caja de la que sale un Android.
 - Maneja los componentes instalables y actualizables del SDK.
- El AVD Manager está disponible a través del entorno de desarrollo:
 - El el botón con el móvil con el Android dentro.
 - Gestiona las máquinas virtuales de Android de nuestro dispositivo.

3.2. SDK Manager (I)



- Al arrancarlo se conecta a internet para actualizar el listado de componentes disponibles.
- Nos permite instalar, actualizar o desinstalar componentes del SDK de Android.



3.2.1. Herramientas/Tools (I)



- SDK tools:
 - Se encarga de la comunicación con el entorno de desarrollo con las máquinas virtuales.
 - Si actualizamos estas “Tools” deberemos actualizar también el plugin del entorno de desarrollo sino no sabremos si podrá funcionar correctamente.

3.2.1. Herramientas/Tools (II)



- Platform-tools:
 - Herramientas de comunicación con los dispositivos de Android, virtuales o no.
 - Permite la actualización de los dispositivos así como pasar o coger archivos de la memoria del dispositivo.
 - Permite también acceder a la *shell* del dispositivo.
- Build Tools:
 - Herramientas de compilación y gestión de los resultados de la misma.
 - Permiten generar los *APK*.

3.2.2. Extras



- Support Repository: Repositorio de soporte.
- Support Library: biblioteca de soporte de versiones antiguas.
- AdMobs ADs SDK: Publicidad.
- Analytics SDK: Analítica de usuarios.
- Play Services: Servicios de la google play.
- Repository: Repositorio oficial.
- APK Expansion: Ampliación de APK, más de 50 MB.
- Billing: Sistema de pagos InApp.
- Licensing: Licencia de seguridad de la google play.
- USB: Driver USB.
- Web Driver.
- Emulador Intel: acelerador de la máquina virtual con micro Intel sobre micro Intel.

Instalamos
todo

3.2.3. APIs



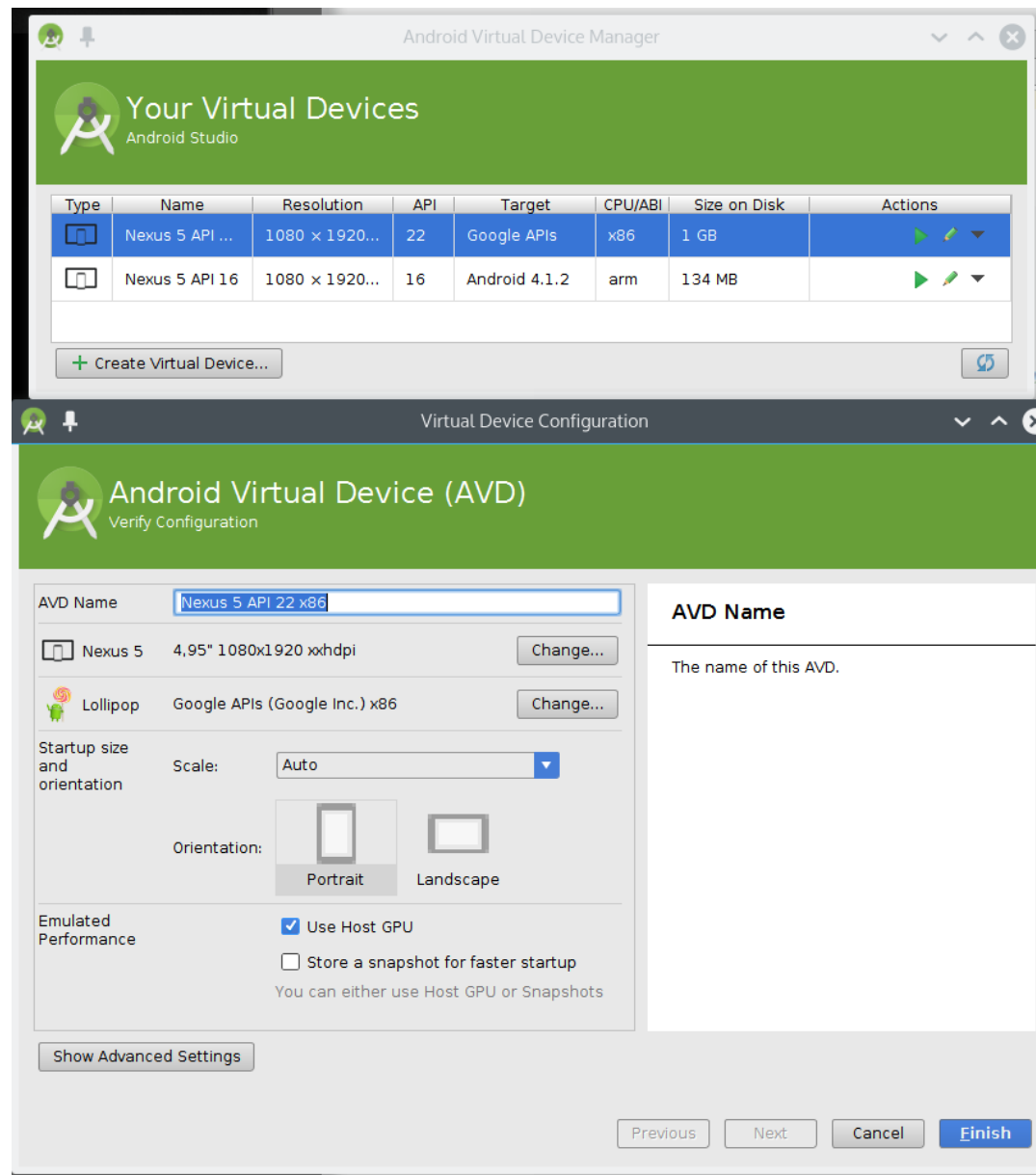
- SDK: Bibliotecas de desarrollo de Android. Imprescindible para compilar.
- Samples for SDK: Ejemplos que podemos utilizar.
- System Images: Imágenes/binarios/ROM que puede utilizar para arrancar la máquina virtual según arquitectura.
- Google APIS (X): Genera máquinas virtual descargando los API's Android, por ejemplo, mapas.
- Sources for Android SDK: Código fuente para resolver dudas o problemas a la hora de manejar alguna clase.
- TV, Glass, Wear...: Para otros dispositivos.

3.3.1. Introducción



- Gestor de máquinas Virtuales de Android.
- Según hayamos descargado las System Image en el SDK Manager así podremos gestionar más o menos máquinas virtuales.
- Al Arrancar veremos las máquinas virtuales que ya tenemos dadas de alta.
- Podemos añadir nuevas pulsando en el botón “New”.

3.3.1. Introducción



3.3.2. Características



- AVD Name: Nombre de la máquina virtual.
- Device: plantillas de dispositivo, como los Nexus.
- Target: versión de Android, con o sin Google API's.
- CPU: Arquitectura, correspondiente a las System Images.
- Keyboard: presencia de teclado físico.
- Skin: Aspecto de la ventana.
- Memoria: RAM y Montículo de la VM.
- Internal storage: memoria interna del dispositivo.
- SDcard: memoria externa del dispositivo virtualizada a un fichero podemos indicar el fichero o el tamaño del mismo.
- Emulation Options: Snap Shot y Use Host GPU (ideal para versiones 4.x).

3.3.3. Tips



- Una vez creada deberá aparecer en el listado.
 - Podremos editarla borrarla, repararla o arrancarla.
- Al arrancar podemos elegir si borrarle la memoria o dejarla intacta.
- Cuando vayamos a ejecutar las aplicaciones podremos elegir si se elige la máquina virtual automáticamente o elegir nosotros manualmente la elegida.



4. Estructura de ficheros

4. Estructura de ficheros



- En la carpeta SDK de nuestro sistema se encuentra todos las plataformas instaladas

