



Android
Studio

TEMA 1.- ANALISIS DE LAS TECNOLOGÍAS PARA MÓVILES

PROGRAMACIÓN MULTIMEDIA y DISPOSITIVOS MÓVILES

Roberto Sánchez de la Rosa
rsanchezro@educa.jcyl.es

CURSO 2025-2026

2º D.A.M

IES RIBERA DE CASTILLA

TEMA 1. - ANALISIS DE LAS TECNOLOGÍAS PARA MÓVILES

1. DISPOSITIVOS MÓVILES: TIPOS, HISTORIA y EVOLUCIÓN
2. SISTEMAS OPERATIVOS DISPOSITIVOS MÓVILES
3. OS ANDROID: UN POCO DE HISTORIA
4. OS ANDROID: ARQUITECTURA
5. OS ANDROID: VERSIONES
6. SOFTWARE PARA DESARROLLO: ANDROID SDK y ANDROID STUDIO

1. DISPOSITIVOS MÓVILES: TIPOS, HISTORIA y EVOLUCIÓN

- NOMBRES PROPIOS

- Nicola Tesla.- Base para la telefonía móvil
- Mónico Sánchez.- Primer móvil
- Martin Cooper.- Primer móvil comercial 1973
- Matti Makkonen .- Inventor del SMS

1. DISPOSITIVOS MÓVILES: TIPOS, HISTORIA y EVOLUCIÓN

- EVOLUCIÓN DE LOS DISPOSITIVOS MÓVILES
 - Video YouTube
 - TIMELINE DISPOSITIVOS MÓVILES



2. SISTEMAS OPERATIVOS DISPOSITIVOS MÓVILES

- Actualmente existen las siguientes plataformas:

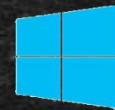
- Apple IOS



- Android



- Widows Phone



Windows
Phone

- Black Berry



- Suscribirse al canal de YouTube de Jesús
- Suscribirse al canal de Aristidevs (ver video de su Historia)
- VER EL VIDEO COMPARATIVA DE PLATAFORMAS (obsoleto)
- Comparativa con otras plataformas

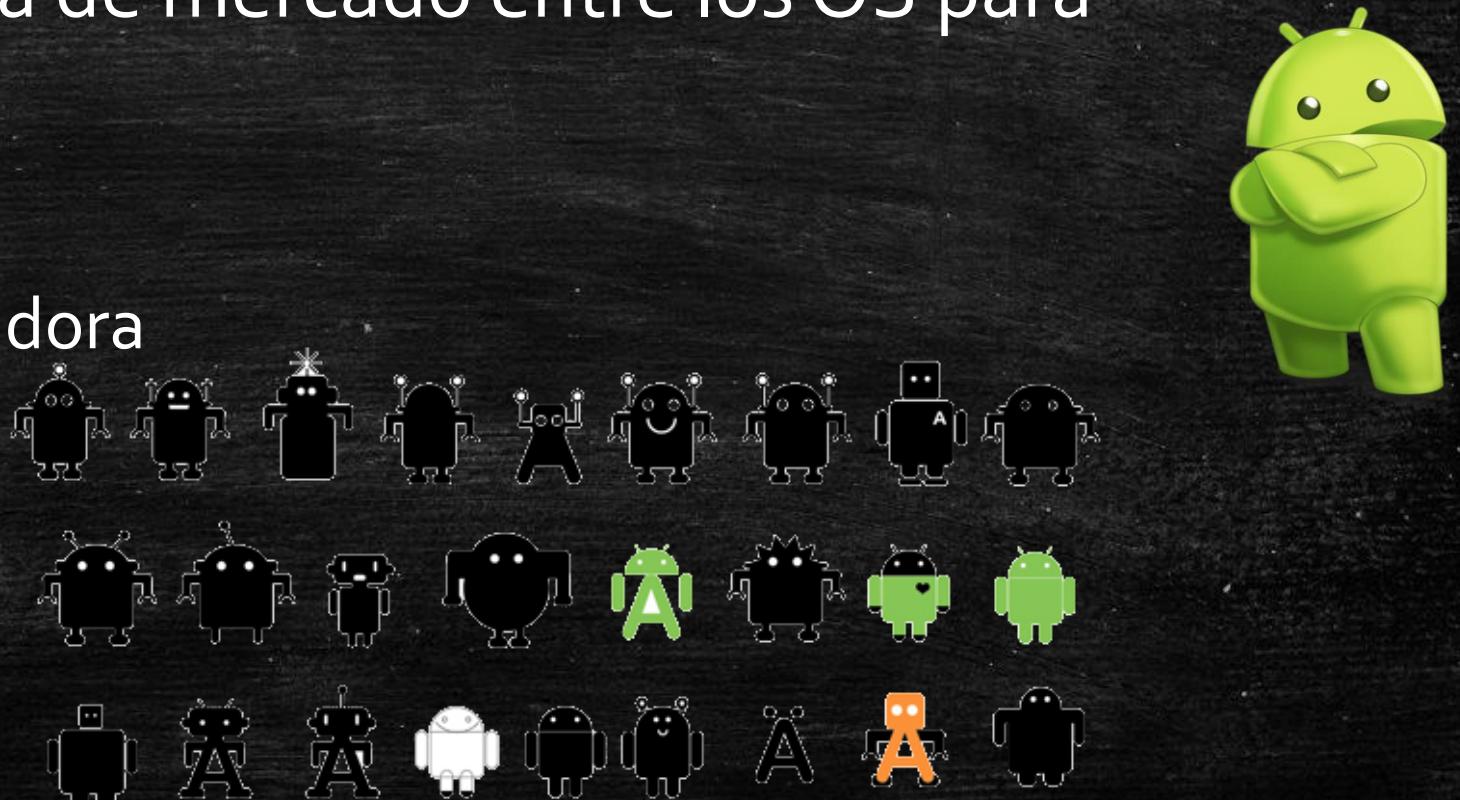
3. OS ANDROID: UN POCO DE HISTORIA

- UN POCO DE HISTORIA
- 2003 Inicios de Android, desarrollado por Android Inc. Y basado en Linux.
- 2005 Google adquiere Android, Inc. Pequeña empresa que desarrolla software para móviles.
- 2007 Nace la Open Handset Alliance. Consorcio de empresas (operadoras, fabricantes, software) unidas con el objetivo de desarrollar estándares abiertos para móviles. Google, Intel, ARM, HTC, LG, Motorola, Samsung, T-Mobile, Vodafone, etc.
- 2008 Se publica Android como Open Source: Licencia Apache 2.0 + otras licencias (GPL v2 para el núcleo). Se abre el Android Market. HTC Dream (G1), primer teléfono con Android.
- 2012 Google Market pasa a ser Google Play Store



3. OS ANDROID: UN POCO DE HISTORIA

- En la actualidad es el OS con más introducción en el mercado, cerca del 80% de cuota de mercado entre los OS para dispositivos móviles.
- LOGO: ANDY
 - Irina block fue la diseñadora
 - Modelos
 - Logo open source



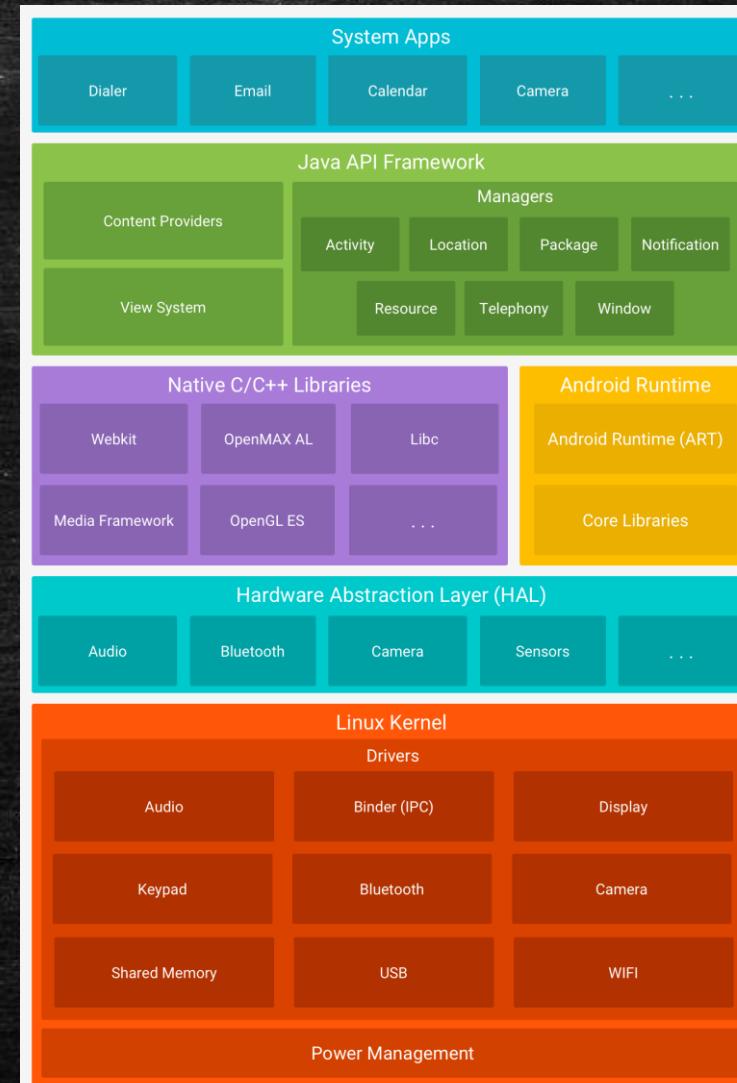
4. OS ANDROID: ARQUITECTURA

■ CARACTERISTICAS

- **PLATAFORMA ABIERTA** basada en Linux y de código abierto
- **ADAPTABLE** a cualquier tipo de hardware: móviles, relojes, tv, autos, cámaras, etc..
=> **ESFUERZO ADICIONAL** al programador, tiene que hacer funcionar su programa en diferentes dispositivos
- **PORTABILIDAD**, el lenguaje de programación Java, aunque en 2013 apareció un lenguaje nuevo ([Kotlin](#)). El concepto de MV lo hace portable
- **ARQUITECTURA basada en INTERNET**. Diseño de interfaz en XML
- **ACEPTABLE NIVEL DE SEGURIDAD**.- Los programas se ejecutan aislados unos de otros ([SandBox en Android](#))
- **OPTIMIZADO** para poca potencia y memoria.- La maquina virtual ART (Dalvik en versiones antiguas). Implementación de google de la JVM, [CONFLICTO Google vs Sun](#)

4. OS ANDROID: ARQUITECTURA

PILA DE SOFTWARE



4. OS ANDROID: ARQUITECTURA

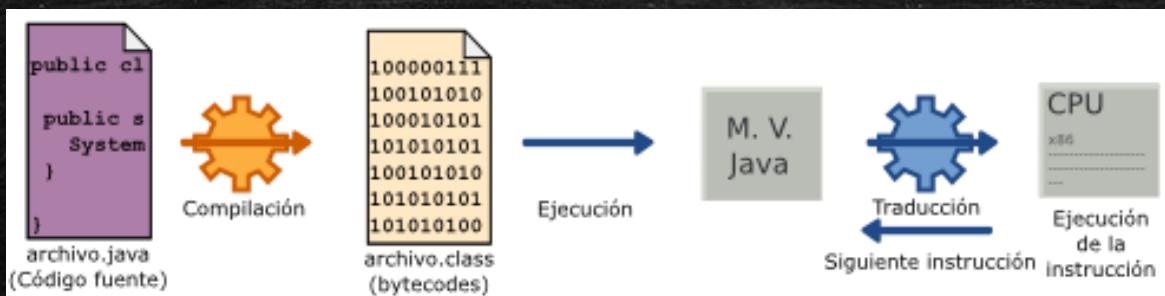
- NUCLEO DE LINUX

- SO LINUX 2.6
- En esta capa se proporciona la seguridad, manejo de memoria, el multiproceso, la pila de protocolos y el soporte de drivers para dispositivos
- Actúa de interface entre el hardware y el resto de la pila.
 - Rootear un sistema ANDROID: Hacerte superusuario
 - Que es el bootloader

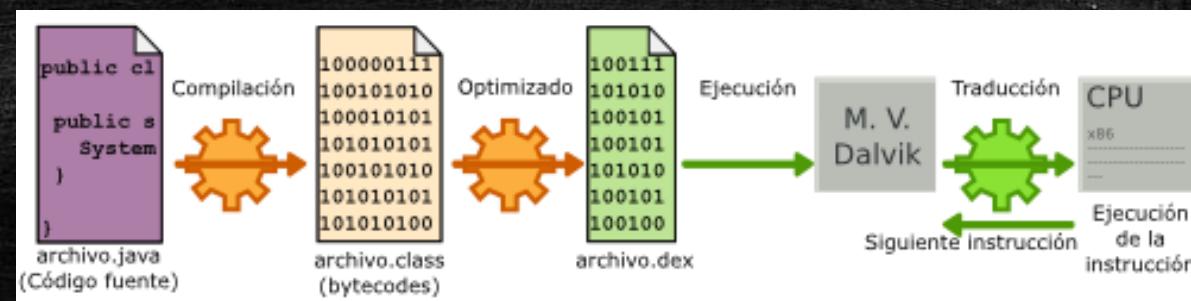
4. OS ANDROID: ARQUITECTURA

RUNTIME DE ANDROID

- Es la Maquina virtual (ART o Dalvik)
- PROCESO DE EJECUCIÓN JAVA



- PROCESO DE EJECUCIÓN ANDROID



4. OS ANDROID: ARQUITECTURA

■ RUNTIME DE ANDROID

- Archivo dex es el ejecutable de la Maquina virtual android (optimizados para ahorrar memoria).
- A partir de Android 5.0 se sustituye DALVIK por ART, optimiza más el uso de CPU y memoria, y por lo tanto de bateria.
- Diferencias entre la arquitectura DALVIK y ART (Muy bien explicado)
- **CORE LIBRARIES**.- Cjto de clases que proporcionan la funcionalidad básica de java.

4. OS ANDROID: ARQUITECTURA

■ LIBRERIAS NATIVAS

- Librerias C/C++ usadas en varios componentes android
- Compiladas en código nativo del procesador
- LISTADO
 - System C Library.- (libc) librería BSD de C estándar
 - Media Framework.- Librería multimedia, soporta codecs para reproducción y grabación audio y video.
 - Surface Manager.- Maneja el acceso al subsistema de representación gráfica 2D y 3D
 - SGL.- Motor gráfico 2D
 - Librerías 3D
 - SQLite.- Motor de bases de datos relacionales
 - SSL.- Servicios de encriptación

4. OS ANDROID: ARQUITECTURA

■ ENTORNO DE APLICACIÓN

- Capa diseñada para reutilización de componentes
- Los servicios más importantes son:
 - VIEWS.- Cjto. De vistas (parte visual de los componentes)
 - Resource Manager.- Acceso a recursos que no son código
 - Activity Manager.- Maneja el ciclo de vida de las aplicaciones, proporciona sistema de navegación entre ellas
 - Notification Manager.- Permite a las apps mostrar alertas personalizadas en la barra de estado
 - Content Providers.- Mecanismo sencillo para acceder a datos de otras aplicaciones (como los contactos)

4. OS ANDROID: ARQUITECTURA

- APLICACIONES

- Capa que contiene las aplicaciones
- Las aplicaciones están escritas en Java, necesario Android SDK para desarrollo
- Hay aplicaciones de más bajo nivel, en C/C++, para ello es necesario Android NDK

- VIDEO ARQUITECTURA

5. OS ANDROID: VERSIONES

- Tres Conceptos:
 - Nombre Comercial.- Nombres de postres en inglés, inicial orden alfabetico
 - Número de versión.- X.Y, donde x implica un cambio importante de versión e y hace referencia a cambios sobre esa versión
 - API Level.- Por cada versión o subversión se añaden o modifica el API, números enteros comenzando desde el 1. Actualmente API Level 36 (Baklava) (Android 16) (APILEVELS)
- Versiones y Distribución Documentación Oficial
- VIDEO y Doc(Versiones_Android.pdf) Sobre versiones

6. SOFTWARE PARA DESARROLLO: ANDROID SDK y ANDROID STUDIO

- ANDROID SDK.- El kit de desarrollo para Android, contiene:
 - Compilador
 - Emulador
 - Depurador
 - Librerías
- ANDROID STUDIO.- Desde Diciembre del 2014 Google ofrece un IDE, que incluye además la instalación del ANDROID SDK
- VERSIÓN Narwhal 3



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- AVD MANAGER.- Android Virtual Device o dispositivo virtual de Android, Necesario para poder ejecutar las aplicaciones emulando un dispositivo o ejecutándolo directamente en un dispositivo real. AVD Manager es el gestor de dispositivos virtuales.
 - [Doc Oficial](#)
 - CARACTERISTICAS
 - Hardware
 - Imagen
 - Área de almacenamiento
 - Mascara

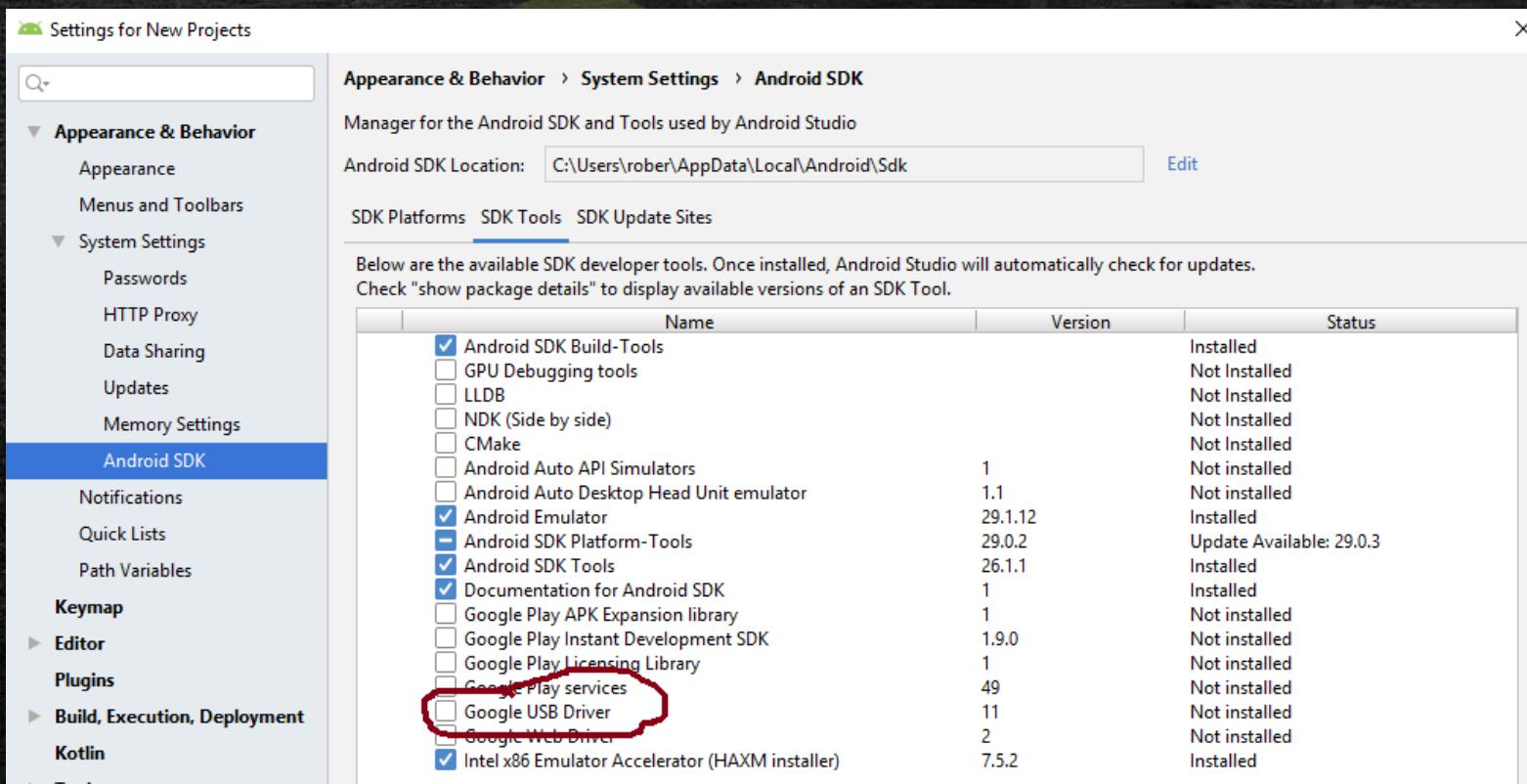
6. SOFTWARE PARA DESARROLLO: ANDROID SDK y ANDROID STUDIO

- Conectar un dispositivo Real y ejecutar la app

- Poner el dispositivo en modo depuración, a partir de la versión 4.4 de android necesario desocultar esta opción en Ajustes>Acerca del Teléfono>Nº de compilación pulsar 7 veces en el número de compilación, Aparece una opción para desarrolladores, habilitar depuración USB
 - Instalar el controlador o bien desde el fabricante o bien usando un controlador genérico => necesario instalar google usb driver desde SDKManager >SDK Tools

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Coneectar un dispositivo Real y ejecutar la app



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Será necesario habilitar la depuración USB

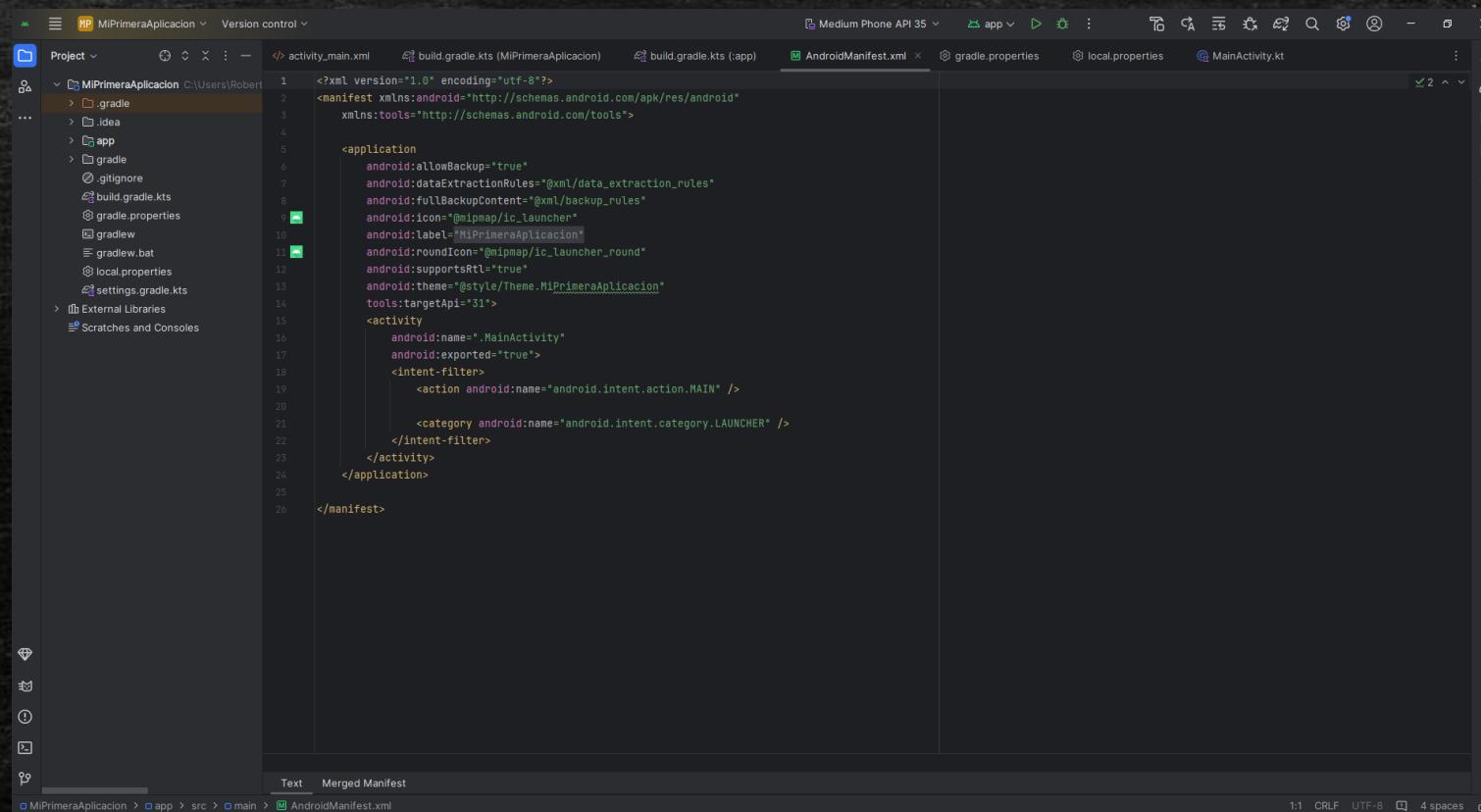


6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Herramienta SCRCPY para visualizar el móvil en el PC
 - Descarga desde Github
 - Ejecución línea comandos: comando scrcpy (opción más robusta)
 - Comando adb para visualizar y configurar dispositivos
- Existen otras herramientas similares
 - LetsView(gratuita)
 - Vysor (semigratuita)

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

■ Introducción a Android Studio (Doc. Oficial)

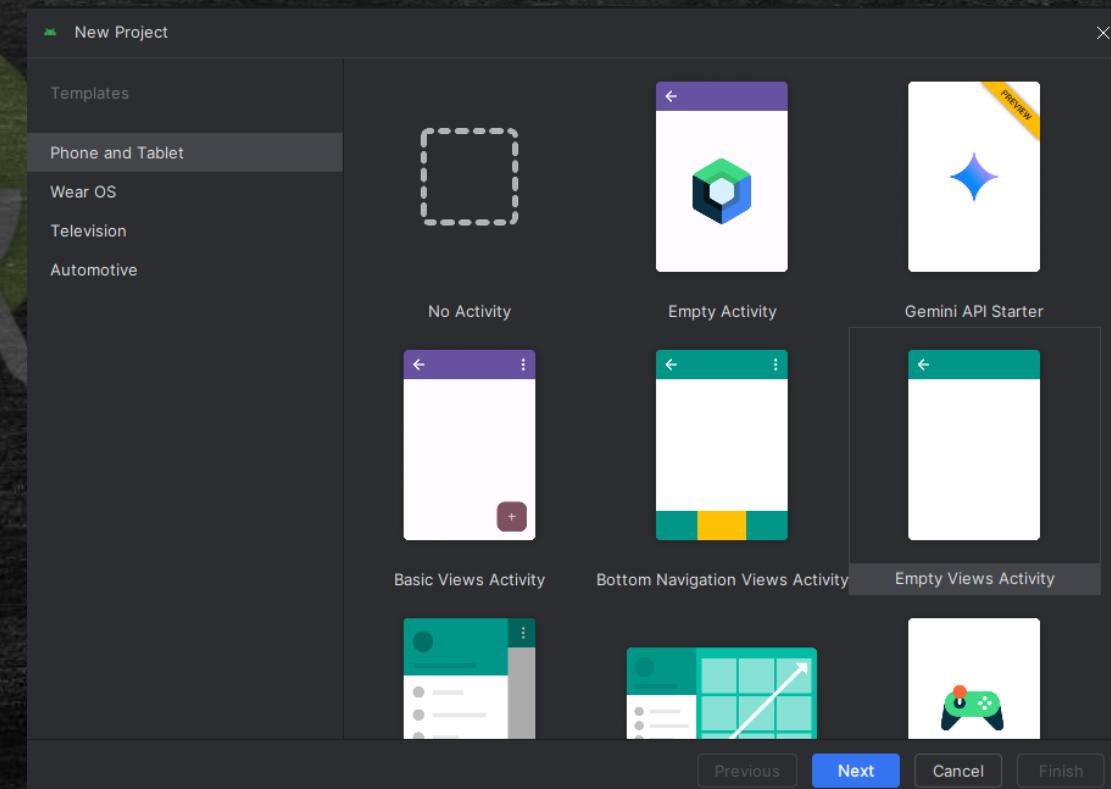


The screenshot shows the Android Studio interface with the following details:

- Project View:** Shows the project structure for "MiPrimeraAplicacion" located at "C:\Users\Robert". It includes files like .gradle, .idea, app, gradle, .gitignore, build.gradle.kts, gradle.properties, gradlew, gradlew.bat, local.properties, settings.gradle.kts, External Libraries, and Scratches and Consoles.
- Main Editor:** Displays the content of the "AndroidManifest.xml" file. The XML code defines a single activity named ".MainActivity" with various attributes and intent filters.
- Toolbars and Status Bar:** Standard Android Studio toolbars and status bar indicating "Medium Phone API 35", "app", and other system information.
- Bottom Navigation:** Shows the current file path: "MiPrimeraAplicacion > app > src > main > AndroidManifest.xml".
- Bottom Status:** Includes "Text" tab, "Merged Manifest" tab, and file statistics: "1:1 CRLF UTF-8 4 spaces".

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

■ Creación de un proyecto



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Creación de un proyecto

- Seleccionar pantalla inicial (Tipo de Activity, por defecto Empty Views Activity)
- Nombre del proyecto.- Significativo de lo que hace
- Nombre del paquete.- Debe ser único, no se pueden publicar paquetes con el mismo nombre (nunca usar example) en play store.
- Localización de proyecto.- IMPORTANTE: No usar espacios en los nombres de directorios.
- Lenguaje de programación.- Java o Kotlin

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Creación de un proyecto (cont.)

- Tres conceptos sobre APILEVEL (archivo build.gradle)
 - minSdkVersion → la versión mínima de Android que tu app soporta.
 - targetSdkVersion → la versión para la que tu app está optimizada (en este caso, API Level 35).
 - compileSdkVersion → la versión que usas para compilar tu app.
- Lenguaje de construcción.- Kotlin DSL (recomendado)

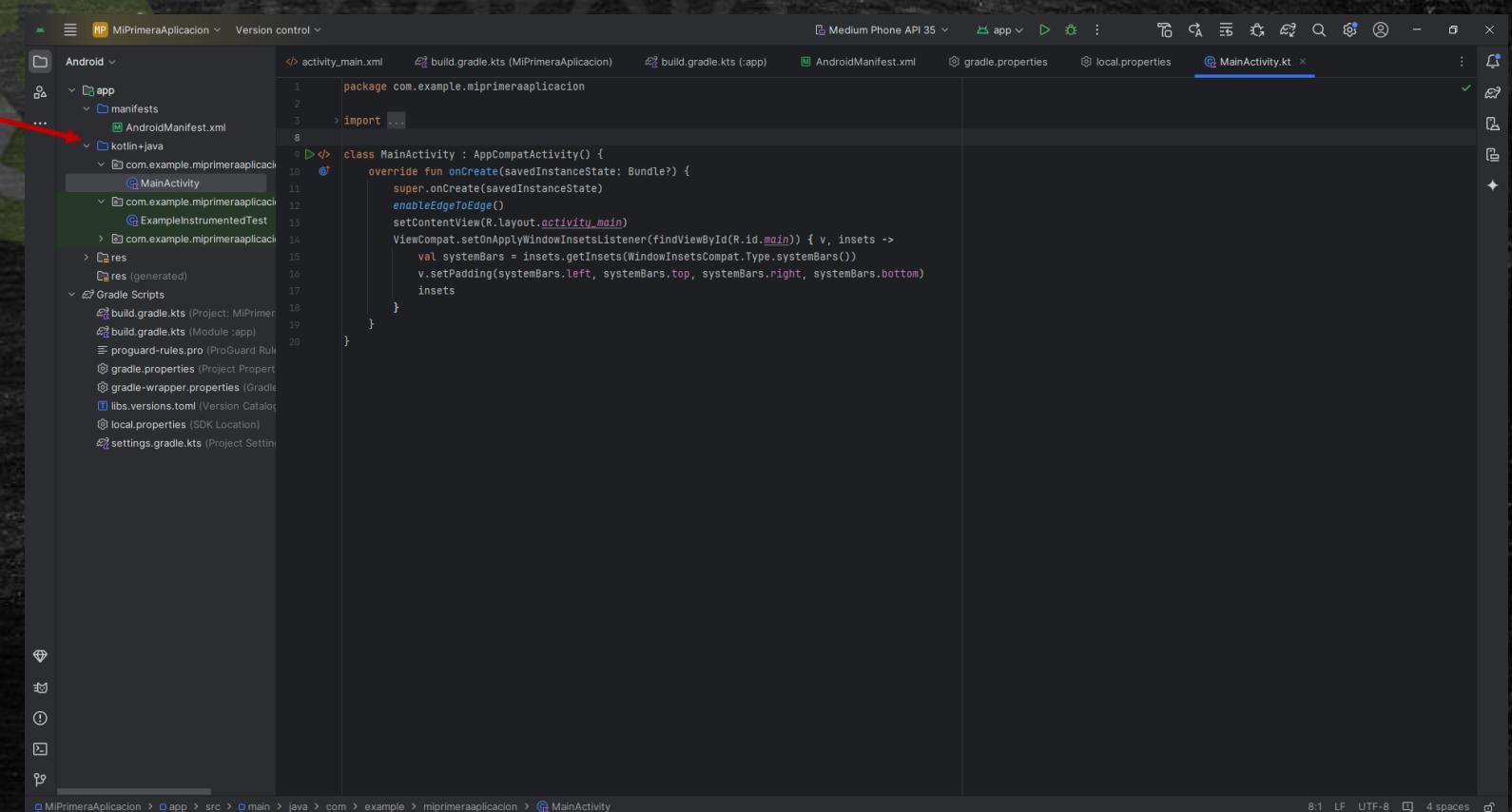
6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Ejemplo 1: Creación de un nuevo Proyecto: HolaMundo
 - EmptyActivity
 - Lenguaje Kotlin
 - Versión mínima la que permita la ejecución en el 95% de los terminales
- Ejemplo 2: HolaMundo2, crear un proyecto con un botón, al pulsarlo saldrá un mensaje personalizado.

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID**
 - Enlace a video explicativo

Visión del
proyecto(Android)

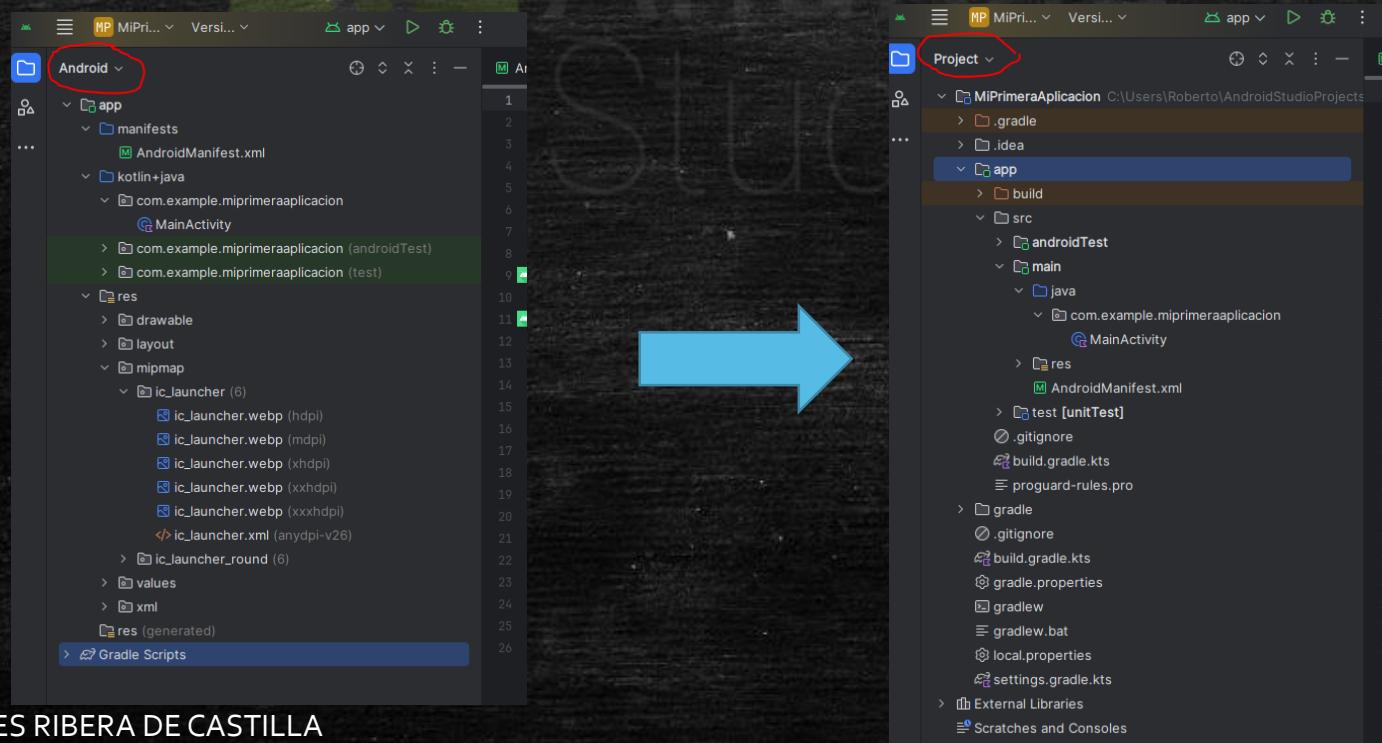


The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The left pane displays the project structure for "MiPrimeraAplicacion". It includes:
 - Android**: A folder containing **app**.
 - app**: A folder containing **AndroidManifest.xml**, **kotlin-Java**, and **res**.
 - kotlin-Java**: Contains **MainActivity.kt** and **ExampleInstrumentedTest.kt**.
 - res**: Contains **layout** and **values**.
 - Gradle Scripts**: Contains **build.gradle.kts**, **build.gradle.kts (Module: app)**, **gradle.properties**, **gradle-wrapper.properties**, **libs.versions.toml**, **local.properties**, and **settings.gradle.kts**.
- Code Editor:** The right pane shows the code for **MainActivity.kt**. The code defines a class **MainActivity** that extends **AppCompatActivity**. It overrides the **onCreate** method to set the content view to **activity_main** and handle window insets.
- Toolbars and Status:** The top bar shows the project name "MiPrimeraAplicacion", the selected API level "Medium Phone API 35", and various tool icons. The bottom status bar indicates "8:1 LF UTF-8 4 spaces".

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID.** - La visión de la estructura de un proyecto se puede ver de varias formas (Android), Project (lo que hay físicamente en las carpetas), hasta 7 perspectivas.



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID.**- VISIÓN

ANDROID: TRES CARPETAS

- manifests.- Incluye el archivo de manifiesto(AndroidManifest.xml)
- Kotlin+java.- Incluye el código fuente en paquetes
- res.- Incluye los recursos

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID:** Archivo de Manifiesto([AndroidManifest.xml](#))
 - en la carpeta manifests, incluye configuración a nivel del módulo de aplicación móvil (carpeta app). Contiene entre otras cosas:
 - Los componentes de la App (activities, services, providers). En cada uno se definen propiedades básicas como el nombre de la clase.
 - Permisos que la aplicación pueda necesitar
 - Funciones software/hardware que requiera la aplicación.

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID:** Archivo de Manifiesto([AndroidManifest.xml](#))

- Etiquetas necesarias: <manifest> y <application>, solo aparecen 1 vez, el resto cero o más veces.
- Convenciones
 - Elementos (etiquetas <>)
 - Atributos, todos comienzan(hay excepciones) con android:
 - Varios valores, se repite el atributo tantas veces como valores
 - Valores de recursos.- @type/name. Donde type es el tipo de recurso(String, drawable,xml) y name identifica el recurso. Ctrl+ClickDercho y abre el recurso

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID:** Archivo de Manifiesto([AndroidManifest.xml](#))

- <application>.- Declara la aplicación, contiene atributos como:
 - allowBackup.- Se incluye dentro de la copia de seguridad del móvil
 - icon.- El icono de la aplicación, enlace a recurso(carpeta) de iconos
 - label.- Etiqueta accesible en toda la aplicación, valor como recurso o String
 - roundIcon.- Iconos redondeados (adaptados), valor como recurso a carpeta
 - theme.- Tema de la aplicación(estilo), dentro del recurso values

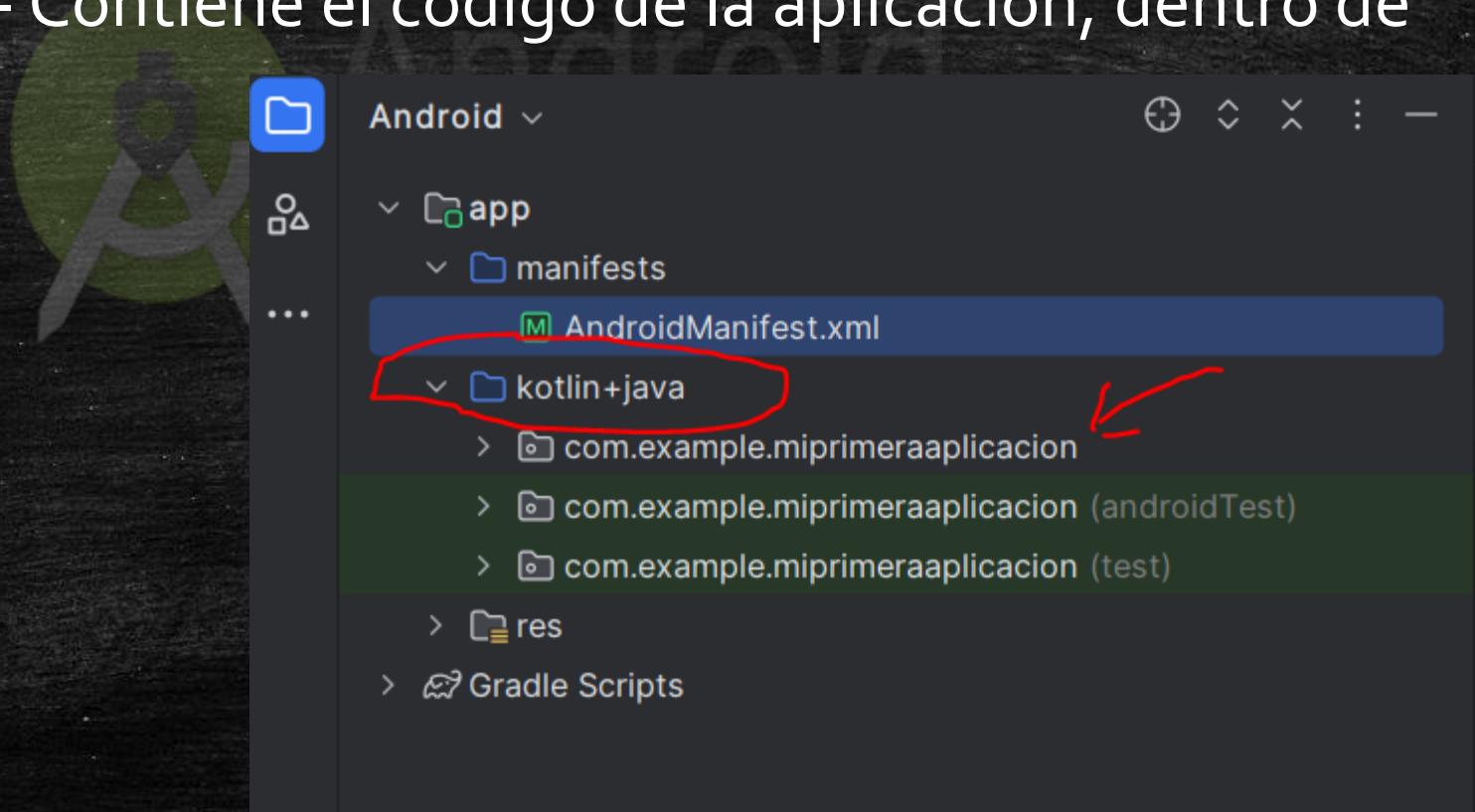
6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID:** Archivo de Manifiesto([AndroidManifest.xml](#))

- **<activity>.**- Dentro de la etiqueta `<application>`, habrá tantas como pantallas tenga la app(activities). Algunos atributos:
 - `android:name`.- Nombre de la clase que implementa la Activity
 - `android:exported`.- permite iniciar la actividad desde otras aplicaciones, necesario si es la actividad principal
- **<intent-filter>.**- Dentro de `<activity>`, especifica los tipos de intents a los que puede responder la activity (o un servicio)

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- ESTRUCTURA DE UN PROYECTO ANDROID ([Doc oficial](#))
 - Carpeta java+kotlin.- Contiene el código de la aplicación, dentro de paquetes.



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

▪ ESTRUCTURA DE UN PROYECTO ANDROID ([Doc oficial](#))

- Carpeta res..- contiene los recursos usados en la aplicación
- **drawable**.- Carpeta que contiene las imágenes y descriptores de imágenes (xml)
- **mipmap**.- Carpeta con el icono de la aplicación, dos tipos de iconos: normal y bordes redondeados, varios ficheros dependiendo de la densidad gráfica del dispositivo: hdpi (alta), mdpi(media), etc.. Decisión en RunTime.
- **layout**.- Contiene ficheros xml con vistas de la app. Nos permiten configurar las pantallas. Interfaces visuales.
- **Menu**.-Contiene xml con los menús de cada actividad (en el primer proyecto no hay menus)

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID** ([Doc oficial](#))

- Carpeta res.- contiene los recursos usados en la aplicación
 - Values.- Ficheros xml para indicar valores usados en la app.
 - Arrays.xml.- Arrays de enteros, strings o recursos
 - Colors.xml.- Colores primarios de la app
 - Dimens.xml.- Margenes de la app
 - Strings.xml.- Cadenas de caracteres de la app
 - Styles.xml.- Estilos y temas de la app
 - Anim.- Carpeta con xml con animaciones de vistas (Tween)
 - Animator.- Carpeta con xml's con animaciones de propiedades
 - Xml.- Otros xml
 - Raw.- Otros ficheros que no estan en formato xml

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

▪ ESTRUCTURA DE UN PROYECTO ANDROID ([Doc oficial](#))

- Fichero build.gradle (dentro de Gradle Scripts).- Guarda las opciones de configuración de compilación del módulo (Module:app) ([Doc. oficial](#))
 - compileSdkVersión.- versión del sdk con la que compilamos la app
 - minSdkVersion.- nivel mínimo del API que requiere la app, si se ejecuta en un sdk antiguo deberás incluir condicionales.
 - targetSdkVersion.- versión más alta con la que se ha puesto a prueba la app
 - versionCode y versionName.- Versión de la app
- **Fichero build.gradle(Proyect)** .- Configuraciones de proyecto
 - repositories.- Hay que indicar el nombre del repositorio para obtener librerías

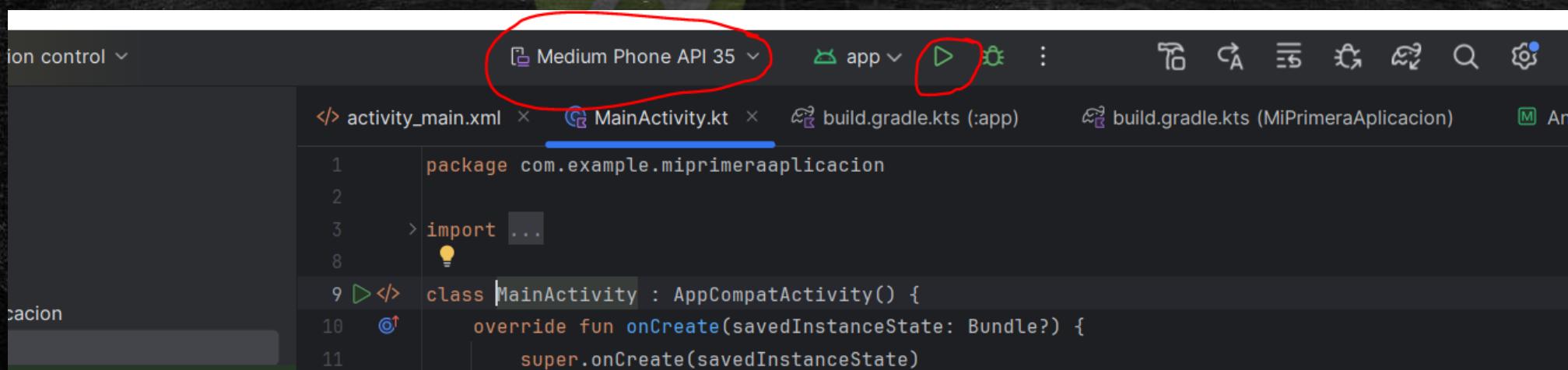
6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- **ESTRUCTURA DE UN PROYECTO ANDROID** ([Doc oficial](#))
 - Fichero build.gradle (dentro de Gradle Scripts).- Cuando hay modificaciones en estos ficheros será necesario realizar sincronización([Sync Now](#))

6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

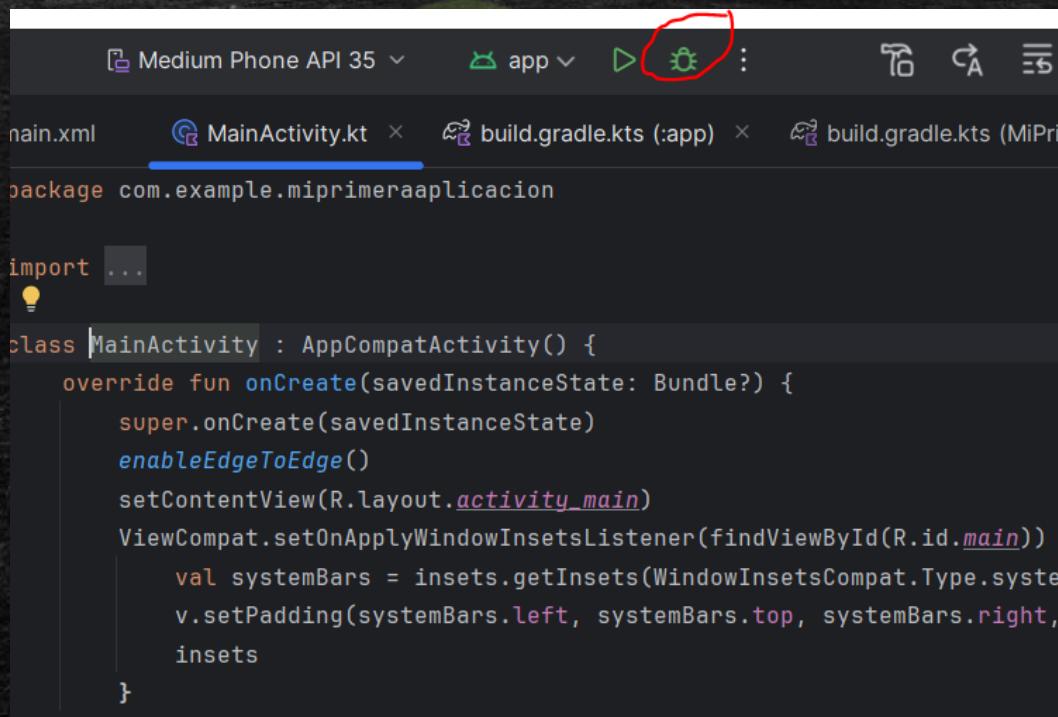
■ EJECUTAR APLICACIONES ([Doc. Oficial](#)).-

- En un emulador
- En un terminal real



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

■ DEPURAR APLICACIONES ([Doc. Oficial](#)).-



6. SOFTWARE PARA DESARROLLO: ANDROID SKD y ANDROID STUDIO

- Ejemplo 1: Creación de un nuevo Proyecto: HolaMundo
 - EmptyActivity
 - Lenguaje Kotlin
 - Versión mínima la que permita la ejecución en el 95% de los terminales
- Ejemplo 2: HolaMundo2, crear un proyecto con un botón, al pulsarlo saldrá un mensaje personalizado.

BIBLIOGRAFÍA / REFERENCIAS

- DOCUMENTACION OFICIAL ANDROID
- DOCUMENTACIÓN OFICIAL KOTLIN
- CURSO ANDROID + KOTLIN
- CURSOS PILDORAS INFORMÁTICAS:
 - Android & Kotlin
 - Android & Java