Desarrollo de aplicaciones móviles I

Tema Nº4:Creando nuestro entorno de desarrollo.

Indicador de logro Nº4:Crea su primera aplicación utilizando el IDE Android Studio.

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº4:**

Creando nuestro entorno de desarrollo.

**Subtema 4.1:**

Inicia a programar para Android: ¿Qué necesito?

**Ejemplos:**

Aunque el desarrollo de librerías de bajo nivel es posible con el Android Native Development Toolkit, vamos a centrarnos en el desarrollo de aplicaciones con el Android Software Development Toolkit.

El SKD de android incluye numerosas y completas API's para facilitar el desarrollo. Algunas de las características más relevantes son:

* Licencias, distribución y desarrollo gratuitos, tampoco hay procesos de aprobación del software.
* Acceso al hardware de WiFi, GPS, Bluetooth y telefonía, permitiendo realizar y recibir llamadas y SMS.
* Control completo de multimedia, incluyendo la cámara y el micrófono.
* APIs para los sensores: acelerómetros y brújula.
* Mensajes entre procesos (IPC).
* Almacenes de datos compartidos, SQLite, acceso a SD Card.
* Aplicaciones y procesos en segundo plano.
* Widgets para la pantalla de inicio (escritorio).
* Integración de los resultados de búsqueda de la aplicación con los del sistema.
* Uso de mapas y sus controles desde las aplicaciones.
* Aceleración gráfica por hardware, incluyendo OpenGL ES 2.0 para los 3D.

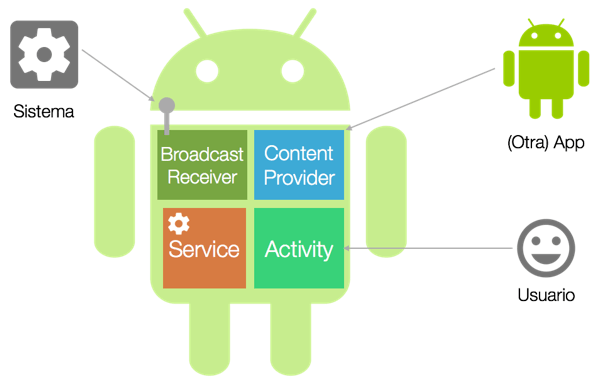
Muchas de estas características ya están, de una manera o de otra, para los SDK de otras plataformas de desarrollo móvil. Las que diferencian a Android del resto son:

* Controles de Google Maps en nuestras aplicaciones
* Procesos y servicios en segundo plano
* Proveedores de contenidos compartidos y comunicación entre procesos
* No diferencia entre aplicaciones nativas y de terceros, todas se crean igual, con el mismo aspecto, y con las mismas posibilidades de usar el harware y las APIs.
* Widgets de escritorio

**Subtema 4.2:**

Arquitectura de una aplicación en Android.

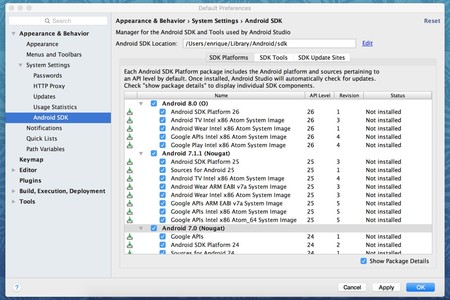
**Ejemplos:**



**Subtema 4.3:**

Android SDK Manager

**Ejemplos:**



Una vez aquí deberemos seleccionar el Android SDK Platform y las APIs de la versión de Android para la que queramos programar. Para quienes quieran probar lo último, les recomendamos descargar la última API disponible (en 2020 es la API level 30). Una vez seleccionados estas versiones y sus archivos, le damos a OK.

Cuando instalamos el SDK de Android también deberíamos seleccionar y descargar herramientas como el **Android SDK Tools, el Android SDK Platform-tools y la última versión del Android SDK Build-tools**, todos ellos imprescindibles cuando queramos instalar ROMs e imágenes de fábrica como las de las preview de Android.

Además de esto, tenemos varios directorios extra relacionados con las API de Google enfocadas a los sistemas operativos Android para accesorios. Nos referimos a Wear OS, Google Cast o Android TV. También cuentan con sus propias API y son muy recomendadas.

**Subtema 4.4:**

Hola Mundo

**Ejemplos:**

Lo primero que debemos hacer para crear el nuevo proyecto es abrir Android Studio y en el menú de la parte superior elegir **File > New > New Project**.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Se nos abrirá la primera ventana del asistente para la creación de un nuevo proyecto.

En esta primera ventana debemos configurar 3 elementos:

* **Nombre de la aplicación**: El nombre que queremos dar a la aplicación.
* **Dominio de la compañía:** Suponiendo que tenemos una página web con un dominio, debemos introducirlo aquí. Si nuestra compañía es www.movilzona.es, el dominio de la compañía a introducir será, solo, movilzona.es.
* **Package name**: El nombre del paquete de java donde guardaremos la aplicación. Android Studio lo generará automáticamente según «dominio.nombre», aunque desde el botón «edit» podemos editarlo.
* **Project Location**: ruta de nuestro ordenador donde vamos a crear el proyecto y donde se almacenarán todos los datos relacionados con él.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Pulsamos sobre «**Next**» y ahora tendremos que elegir tanto el tipo de dispositivo para el que vamos a crear la aplicación como el SDK que vamos a utilizar para ello.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En nuestro caso la mayoría de los proyectos van a ser para smartphones y tablets, por lo que marcaremos solo la primera entrada. Igualmente debemos elegir el «**SDK mínimo** «, es decir, la versión de Android mínima necesaria para que la aplicación funcione correctamente. Por ejemplo, si elegimos el SDK API19, nuestra aplicación funcionará sin problemas en Android 4.4 y todas las posteriores, aunque no lo hará en Android 4.3 ni en las anteriores.

Es decir, a la hora de elegir el SDK mínimo debemos elegir el más antiguo posible (dentro de las posibilidades de nuestro proyecto) para que nuestra aplicación funcione en el mayor número de dispositivos posibles y pueda llegar a un mayor número de usuarios.

Si, por ejemplo, elegimos el **SDK API 23**, la aplicación solo funcionará para aquellos que ejecuten Android 6.0 o posteriores, es decir, para menos del 1% de los dispositivos del mercado, tal como nos indica Android Studio.

Usar el **SDK API 16** nos garantiza compatibilidad con más del 92% de los dispositivos del mercado.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Pulsamos sobre Next y el asistente nos preguntará el tipo de actividad que vamos a crear. Por defecto, como no necesitamos funciones avanzadas, vamos a elegir una «**Blank Activity**«.

Interfaz de usuario gráfica, Aplicación, Teams

Descripción generada automáticamente

Pulsamos sobre «**Next**» de nuevo y debemos configurar el nombre y el tipo de la actividad.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Desde aquí también desde aquí podemos indicarla que la actividad funciones como un fragmento, concepto visto en las definiciones. Por defecto, para no empezar complicando demasiado el desarrollo, dejaremos estos valores de momento por defecto. Más adelante, según nazca la necesidad, veremos cómo modificarlos para facilitarnos el desarrollo.

Todo listo. Pulsamos sobre «**Finish**» y Android Studio comenzará a preparar nuestra aplicación.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Una vez lista veremos la ventana principal de la aplicación, a partir de la cual empezaremos ya con el desarrollo de nuestra aplicación, tal como veremos en las próximas guías.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

[**MI PRIMERA APLICACIÓN EN ANDROID STUDIO ¡HOLA MUNDO!**](https://www.movilzona.es/tutoriales/android/desarrollo/curso-de-desarrollo-android-tema-6-mi-primera-aplicacion-en-android-studio-hola-mundo/)

Lo primero que debemos hacer es abrir Android Studio y crear un nuevo proyecto. Este proceso ya lo hemos explicado así que continuemos.

Lo primero de lo que nos daremos cuenta es que cuando creamos un nuevo proyecto para Android tendremos ya, por defecto, ¡un proyecto «Hello World!» básico. Lo podemos ver porque en la pre visualización de la derecha del código así nos aparece. Nosotros vamos a repasar todos los aspectos fundamentales de este sencillo, básico y elemental programa para poder comprender el por qué esto es así y, a grandes rasgos, cómo funciona la programación para Android.

En nuestro IDE, en la parte izquierda, podemos ver todos los archivos de código que forman nuestra aplicación para Android en forma de un árbol de directorios, así como los scripts de Gradle, en los cuales no vamos a profundizar (de momento).

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Si hacemos doble clic sobre cualquiera de los ficheros podemos ver su código (aunque si el archivo es un fichero xml probablemente lo que seamos será la «previsualización», por lo que debemos mirar en la parte inferior central de la ventana, donde veremos dos pestañas: Design y Text. Pulsamos Text).

El primer código que vamos a ver es el fichero Java «MainActivity» (creado durante el asistente inicial como la actividad básica) que se encuentra en la ruta app>Java>com.example.nombredelaapp>MainActivity.

Interfaz de usuario gráfica, Texto, Aplicación, Word

Descripción generada automáticamente

Si abrimos el código fuente de este fichero podremos ver una definición básica de esta actividad. Lo primero que debemos tener en cuenta (aunque lo analizaremos más adelante) es que nuestra ventana principal, llama MainActivity se extiende de la clase AppCompatActivity.

Dentro de esta ventana, o actividad, podremos encontrar la definición de un archivo de recursos xml, que será el que contenga lo que se va a ver en la pantalla, es decir, en nuestro caso el texto y las propiedades del «Hola Mundo!» que queremos imprimir en pantalla.

Por lo tanto, ¡para poder crear un elemento «Hola Mundo!» en esta sencilla aplicación debemos ir al archivo ahí definido, es decir, a la ruta app>res>layout>activity\_main.xml.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Los usuarios que ya dominen el lenguaje Java sabrán que la instrucción necesaria para mostrar el Hola Mundo! en el lenguaje Java es System.out.println(«Hola Mundo!»), sin embargo, para introducir texto (y otros elementos que veremos más adelante) en una aplicación para Android debemos hacerlo a partir de ficheros xml, a los cuales debemos llamar en la actividad en cuestión para que se muestren en la pantalla. Esto facilita el desarrollo de la aplicación y permite, por ejemplo, hacer de una forma muy sencilla que nuestra aplicación sea multi-idioma.

Android Studio puede funcionar como un IDE D&D (drag and drop) que nos permiten colocar ciertos elementos en el programa arrastrándolos desde la lista (profundizaremos en ellos más adelante) sin tener que programarlos uno a uno.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Para ver una prueba de este sencillo «Hola Mundo!», desde la pestaña «Design» que hemos visto antes pulsaremos en el apartado de «previsualización» sobre el «Hello World!» y lo eliminaremos con la tecla supr. El cuadro de texto que se crea por defecto se eliminará y tendremos un proyecto totalmente vacío.

A continuación, desde la lista de elementos que nos aparece en el centro de la ventana seleccionaremos «Widgets > Large text» y lo arrastraremos a la imagen del smartphone que nos aparece a la derecha, a la posición que deseemos. El resultado final deberá ser similar al siguiente.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

En la parte derecha de la pantalla veremos la sección de «propiedades» del cuadro de texto que acabamos de crear. Desde aquí configuraremos la mayoría de los aspectos del elemento, entre otros, el contenido de este cuadro de texto (y el color, tamaño, etc en caso de que lo necesitemos). Profundizaremos en esto en futuras lecciones.

Dentro de las propiedades, buscaremos el apartado «Text» y cambiaremos el contenido que pone por defecto (Large Text) por el texto que queremos escribir, en nuestro caso, «Hola Mundo!». El aspecto final será similar al siguiente.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**Cómo probar nuestra aplicación «Hola Mundo!» en Android Studio**

Nuestro primer programa está listo. Ya podemos ejecutar nuestro primer programa en la máquina virtual. Para ello pulsaremos sobre el botón «Run App» (o pulsaremos Alt+F10) y esperaremos a que nuestra primera aplicación cargue en la máquina virtual.

Interfaz de usuario gráfica, Aplicación, Word

Descripción generada automáticamente

Captura de pantalla de un celular

Descripción generada automáticamente

**Subtema 4.5:**

Creando un emulador

**Ejemplos:**

Todo empieza desde Android Studio, en donde debes tocar en el pequeño botón del **AVD Manager**. AVD son las siglas de Android Virtual Device, o dispositivo virtual de Android. Si no lo encuentras, también se encuentra en el menú Tools

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

La primera pantalla de bienvenida no dice gran cosa cuando no hay ningún dispositivo creado. Tiene literalmente un botón, que es justo el que debes pulsar para comenzar. Pulsa **Create Virtual Device**, o crear dispositivo virtual.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ahora viene la parte interesante: **elegir qué dispositivo estás emulando**. Esto se usa para usar características propias de dicho dispositivo, especialmente la resolución y tamaño de su pantalla. Algunos de ellos incluyen su propia "piel" que hace que el emulador se vista como ellos.

Puedes elegir cualquiera, aunque **aquellos que incluyen el icono de Play Store** te permitirán usar la tienda de Google, si descargas la imagen del sistema apropiada. Cuando lo tengas, pulsa *Siguiente*.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## 2. Descarga la imagen del sistema

El anterior paso ha sido rápido, pero ahora viene el momento de poner a trabajar el router y PC. Necesitas **elegir la imagen del sistema** que vas a instalar en el dispositivo virtual. Normalmente no tendrás descargada ninguna, así que eso es lo primero que necesitas hacer, descargarla.

Puedes elegir cualquiera, aunque es mejor que te ciñas a la primera pestaña, las **recomendadas**. Éstas están basadas en la misma arquitectura que tu procesador, de modo que funcionan muchísimo más rápidas que las que no. Solo aquellas que lo especifican incluyen Google Play, otras incluyen los servicios de Google pero no Google Play, y otras ninguno de ellos.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Antes de descargar nada necesitas aceptar los **acuerdos de licencia** para el uso de la imagen del sistema y el SDK de Android que estás a punto de descargar. La descarga no empieza hasta que lo hagas.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Tomará un tiempo, pues las imágenes del sistema ocupan varios cientos de megas, pero en cuanto termine de descargarse e instalarse volverás a la pantalla de antes. La diferencia es que ahora la imagen de sistema que descargaste **puede seleccionarse**. Hazlo y luego pulsa *Next*.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## 3. Carga el emulador

A estas alturas ya has hecho todo el trabajo y lo único que te falta es disfrutar del resultado. La misma ventana que antes no mostraba nada incluye ahora tu nuevo dispositivo virtual. Haz **clic en el botón de reproducción** y, si todo ha ido bien, se lanzará el emulador.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Android puede tardar un buen rato en aparecer, según la potencia bruta de tu ordenador, pero antes o después deberías ver **Android metido en un teléfono virtual en tu escritorio**. Con un PC potente, las últimas versiones del emulador van igual de fluidas que un dispositivo real.

Imagen de la pantalla de un computador

Descripción generada automáticamente con confianza media

**Actividad:**

1. CUESTIONARIO TÉCNICO

* ¿Cuál es el concepto de activity en el contexto del desarrollo de Aplicaciones móviles en Android?
* ¿Qué consideraciones debemos de tomar al elegir el SDK Mínimo para nuestro proyecto?
* ¿Qué papel juegan los archivos XML en el desarrollo de Aplicaciones móviles en Android Studio?
* Indique dónde se configura la aceleración de hardware de Intel VT-X o AMD-V para no tener problemas al instalar un emulador
* ¿A qué se le denomina un IDE Drag and Drop?

1. CONCLUSIONES DE LA EXPERIENCIA

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_