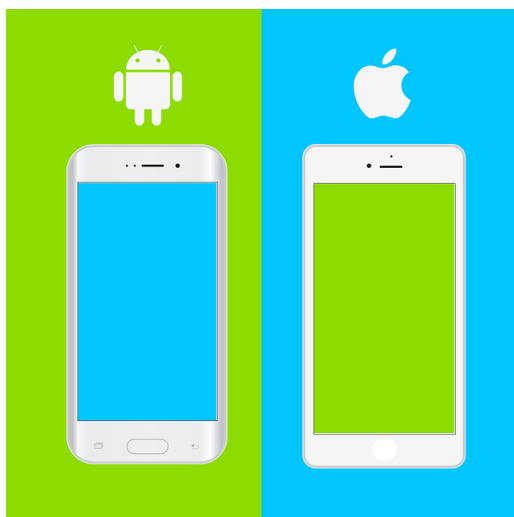


ANÁLISIS DE TECNOLOGÍAS PARA APLICACIONES DE LOS DISPOSITIVOS MÓVILES.

*Programación multimedia y dispositivos móviles.
(PGL)*



U.T. N°1

INTRODUCCIÓN

►Objetivos.

- Dispositivos móviles: tipos, características y limitaciones.
- Plataformas y sistemas operativos: Entornos integrados de desarrollo, emuladores, lenguajes.

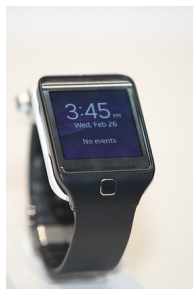


DISPOSITIVOS MÓVILES

- El avance de la tecnología y el uso de los medios inalámbricos ha hecho que en la actualidad, la comunicación sea posible en cualquier momento y desde cualquier lugar .
- Desde los teléfonos de grandes dimensiones, muy costosos y con baterías que duraban apenas 60 minutos hemos pasado a disponer de una amplia variedad de dispositivos con precios muy asequibles, baterías que duran más de 12h. Y con acceso a internet a alta velocidad...

Dispositivos.

- *Objetos personales inteligentes.*
- *Accionados por sensor.*



Teléfonos.

- *Teléfonos inteligentes.*



Ordenadores.

- *Portátil.*
- *Tablet.*



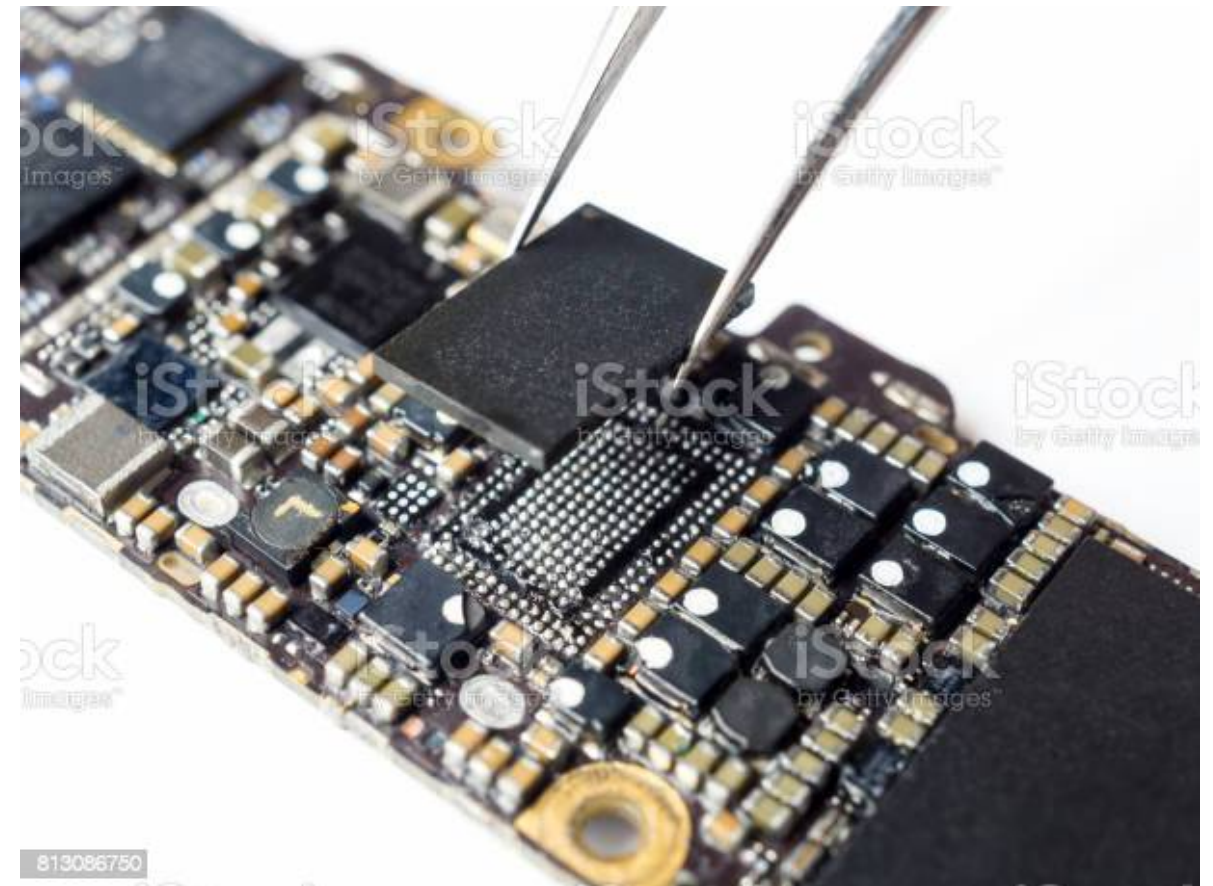
DISPOSITIVOS MÓVILES

- ¿Todo son ventajas?.
- No.El desarrollo de dispositivos trae consigo el problema de la integración entre diferentes plataformas, lo que obliga a la definición de estándares y protocolos.
- Características y limitaciones.Una aplicación móvil NO es una aplicación de escritorio adaptada a un dispositivo de pantalla pequeña sino que deben ser aplicaciones completamente diferentes por las siguientes razones:
 - La capacidad para comunicarse desde cualquier lugar cambia la forma de interactuar con el usuario.
 - La interfaz para comunicarse con los usuarios es diferentes.
 - Los canales de información son diferentes: voz, geolocalización, mensajería, etc..
 - La capacidad de las redes inalámbricas. Redes de datos, etc.



DISPOSITIVOS MÓVILES

- Limitaciones:
 - Relacionadas principalmente por el hardware y la capacidad de conexión de cada dispositivo móvil.
- Procesadores y memoria:
 - Han ido mejorando con el tiempo pero debemos evitar los desarrollos con sobrecarga de elementos multimedia que requieren ciclos muy largos.



DISPOSITIVOS MÓVILES

➤ Pantallas:

- El tamaño y la iluminación son muy importantes en el diseño de la aplicación.
- Debemos facilitar al usuario la entrada de datos, evitando la escritura en campos de texto y facilitando la navegación por botones y la selección mediante despleables, por ejemplo.
- Es fundamental la distribución de los elementos en la interfaz y cómo se navega entre ellos, intentando que sea intuitiva.

➤ Accesos a internet:

- Hay que tener en cuenta que se pagan diferentes tarifas por bytes descargados (aunque las compañías ya ofrecen tarifas sin limitación de datos) y no por tiempo. De, y estos tiempos deben ser aceptables y mantener siempre al usuario informado del tiempo estimado de la operación que esté realizando.

DISPOSITIVOS MÓVILES

- Debemos tener en cuenta:
- **Principal función** de un teléfono móvil es la **realización/recepción de llamadas**, con la máxima prioridad. De forma que habría que mantener el estado de la aplicación cuando se produce la interrupción y poder volver luego al mismo punto.
- **Conexiones**: no sabemos si existirá, si habrá buena cobertura, etc... por lo que hay que comprobar el funcionamiento de la aplicación en todas las situaciones.
- **Emuladores**: Se suelen utilizar para comprobar el comportamiento de la aplicación en diferentes situaciones, pero suelen estar instalados en equipos más potentes que el dispositivo, por lo que SIEMPRE será necesario probarlo en entorno real (en el dispositivo para el que ha sido diseñado) antes de la distribución.

TECNOLOGÍAS

- ANDROID.
- Es un sistema operativo basado en Linux creado por Google para sus dispositivos. Podemos decir que su éxito se basa en su interfaz de usuario basada en tecnología DMI (Direct Manipulation Interface), interfaz con representación de objetos gráficos, fáciles de interpretar y usar y con feedback. Diseñado para dispositivos con pantallas táctiles principalmente, smartphones, tablets, smart tv, cámaras digitales, Android Auto y otros.
- Está programado en C/C++ con licencia opensource (combinación software libre y propietario).
- Cada versión de Android tiene un nombre-clave y un nivel de API que especifica un conjunto de librerías que se utilizan para desarrollar una aplicación.



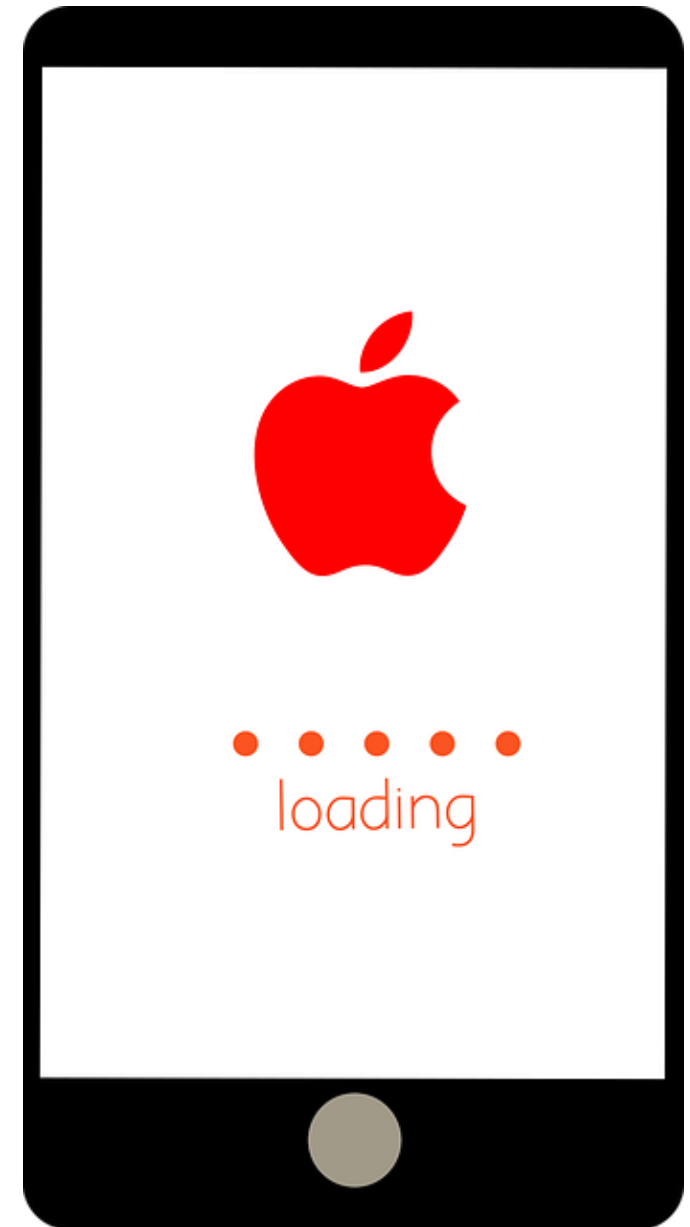
TECNOLOGÍAS

- Uno de los principales problemas del desarrollo Android será programar dando soporte al mayor número de API's
- Ejemplos versión 4.1, Jelly Bean o 5.0, Lollipop que incorpora soporte 64 bits.
- El SDK de Android (Software Development Kit) contiene las herramientas para desarrollar aplicaciones y estarán desarrolladas en Java. En Eclipse se precisa plugin.



TECNOLOGÍAS

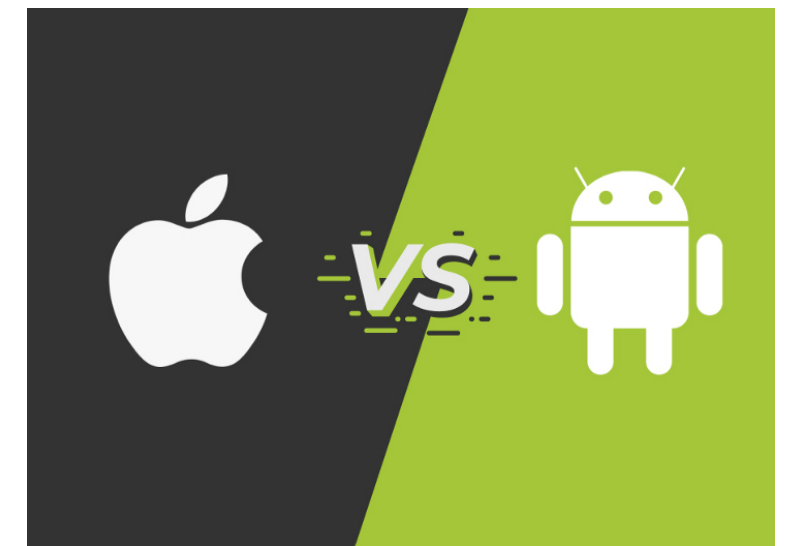
- IOS
- Tecnología de Apple, inicialmente llamada Iphone OS.
- IOS no permite Java y esto hace que algunos componentes de algunas páginas no sean accesibles desde sus navegadores.
- Sólo se puede instalar en dispositivos Apple.
- La arquitectura IOS se basa en capas. Las capas más altas contienen los servicios y las tecnologías para el desarrollo de aplicaciones y las capas más bajas controlan los servicios básicos.
- El IDE para IOS es XCode + SDK. XCode es gratuito pero los desarrolladores deben pagar una cuota por la venta de aplicaciones.



TECNOLOGÍAS

➤ Diferencias:

- Los dispositivos de Apple suelen ser más caros. Y Android puede instalarse en dispositivos más económicos de una amplia variedad de fabricantes.
- Android permita la instalación de muchas aplicaciones gratuitas.
- Respecto a las aplicaciones existentes, las importantes están disponibles para ambos sistemas, aunque suelen aparecer primero en IOS. Si se trata de aplicaciones o juegos de calidad, suelen aparecer en IOS y no estar disponibles en Android. ¿Por qué? Puede ser que el desarrollo para IOS es más lucrativo para los desarrolladores.
- Para la descarga de aplicaciones tenemos Play Store de Google o App Store de Apple. Apple no permite descarga de terceros, en cambio para instalar otras aplicaciones en Android basta con descargarse una apk.



TECNOLOGÍAS



- Personalización del móvil, Android ofrece más libertad que IOS, permite establecer temas, añadir widgets o incorporar aplicaciones de terceros al inicio.
- Por otro lado, tanto las restricciones a la hora de instalar otros programas como un control de actualizaciones más riguroso, hacen a IOS una plataforma más segura. Su protección contra malware es mayor y su firmware más robusto.
- IOS es más simple y fácil de usar además de ser más uniforme para todos los dispositivos IOS, mientras que Android puede variar según el fabricante.
- Y entre otras diferencias, se puede concluir que Android es más abierto y IOS más seguro.

LENGUAJES DE PROGRAMACIÓN



Java

- Es es lenguaje nativo que usa Android
- La JVM permite la *conversión* para Android e IOS



Kotlin

- Inspirado en Java, muy simple y sencillo
- Compatible también con IOS



Swift

Lenguaje para IOS



Objective-C

Para IOS, precursor de Swift



C++

- Para Android, pero no recomendado.
- Desventaja: más complicado y no libera memoria



Dart

Flutter

Flutter es un framework que usa como lenguaje de programación Dart. Que es un lenguaje creado por Google pensado para trabajar desde el cliente, que hoy se usa casi únicamente en conjunto con Flutter. Ahora hay incertidumbre sobre que va a hacer Google con el Proyecto Flutter.

IDES



- Se pueden utilizar entornos de desarrollo generales (Eclipse, Netbeans...), instalando el plugin correspondiente o bien utilizar IDEs específicos. Los IDE incorporan emuladores para la simulación de las aplicaciones sin necesidad de utilizar un móvil real durante el desarrollo y las pruebas. Vemos diferentes casos:
- **Android:**
- Las herramientas las ofrece el SDK. Se puede utilizar un IDE como Eclipse u otro e instalar el plugin ADT (Android development tool).
- Incorpora un emulador que utiliza configuraciones AVD (Android Virtual Device, Dispositivo Virtual para Android). El AVD permite definir las características hardware del teléfono a emular.
- Android Studio es el IDE oficial de Android y reemplazó a Eclipse.

IDES



- iOS:
- Necesita también instalar el SDK de Apple, que incluye el Xcode.
- El emulador iOS permite comprobar el funcionamiento de la aplicación y puede:
- Ofrece funcionalidades como:
- rotar, cuando gira el dispositivo la aplicación rota con él
- cambiar de dispositivo
- comprobar el funcionamiento de la aplicación para diferentes versiones del
- iOS
- o Etc...