**Daniel Espinosa Garcia**

**Proyecto RA1 RA2 - Primera Fase con los CE**

**RA1**

c) Se han identificado las tecnologías de desarrollo de aplicaciones para dispositivos móviles.

Como podemos observar a lo largo del tiempo han existido múltiples sistemas operativos para los cuales se desarrollaron diferentes tecnologías para desarrollar aplicaciones.

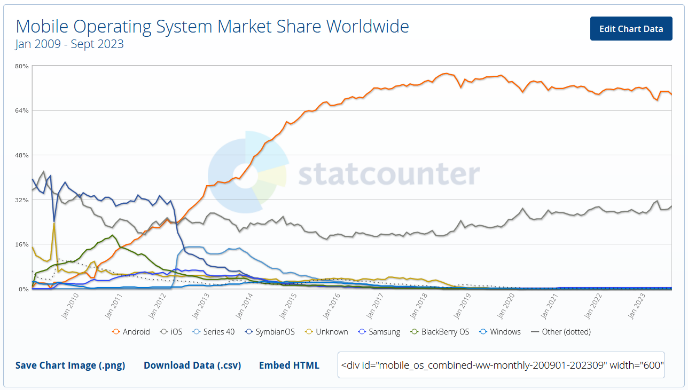


Ilustración https://gs.statcounter.com/os-market-share/mobile/worldwide

En la actualidad el mercado esta dominado por los sistemas operativos de Android y iOS.

Estos dos sistemas operativos (SO) utilizan tecnologías distintas para la programación de aplicaciones en el lenguaje nativo de cada SO

**Swift**: Para desarrollo en iOS, se utiliza este lenguaje de programación ofrecido por Apple para el desarrollo de aplicaciones que funcionen en sus plataformas: iOS, macOS, tvOS, watchOS y iPadOS.

**Kotlin:** Para desarrollo en Android, este lenguaje permite crear apps nativas de alta calidad y gran rendimiento en todas las plataformas soportadas por Android: smartphones, televisores, relojes, etc.

La principal desventaja de estos dos sistemas es que el desarrollo de aplicaciones en los lenguajes nativos no permite portar las aplicaciones entre sistemas operativos por lo cual se requiere duplicar el trabajo.

Para evitar esta duplicidad de trabajo se desarrollaron otras tecnologías para poder portar a ambos SO

**Ionic:** En primer lugar, tenemos ionic, el framework probablemente más popular que nos permite desarrollar aplicaciones multiplataforma. Nos ayuda a construir apps con HTML, CSS3 y Javascript empleando todo el potencial de las tecnologías de desarrollo web.

**Flutter:** Seguidamente, encontramos Flutter, una herramienta desarrollada por Google para el desarrollo de aplicaciones multiplataforma. Está basada en Dart y aunque se presentó en 2015, no se lanzó hasta 2018. Una gran ventaja frente a otras soluciones es que la aplicación desarrollada puede funcionar hasta en 4 plataformas: iOS, Android, Web y Desktop.

**React Native:** Es un framework creado por Facebook en 2015 para el desarrollo de apps multiplataforma.

Al igual que Ionic, también utiliza Javascript, pero lo hace bajo el pretexto de orquestar controles de interfaz de usuario (UI) específicos de la plataforma.

d) Se han instalado, configurado y utilizado entornos de trabajo para el desarrollo de aplicaciones para dispositivos móviles.

Para desarrollar aplicaciones móviles instalamos Unity y Android Studio ambos IDE permiten la creación de apps.

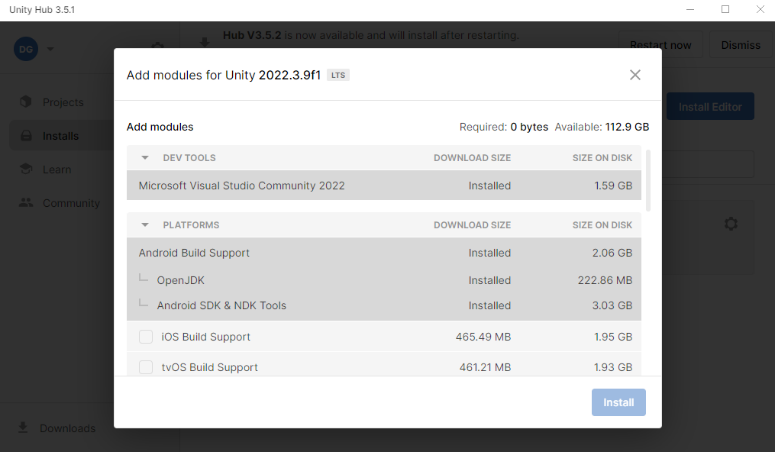


Ilustración configuración de Unity Hub para permitir la exportación a SO Android

Algunos de los IDEs más populares son:

**Android Studio**: es una popular herramienta de programación recomendada por Google. Viene con características de alto nivel para facilitar el trabajo de los desarrolladores en el desarrollo de aplicaciones Android. Una de las características esenciales son sus dispositivos virtuales.

**Xcode**: es un IDE de uso común para los sistemas operativos macOS. Utilizando este IDE, los desarrolladores pueden desarrollar software o aplicaciones en Mac que pueden utilizarse en iOS, iPadOS, macOS, tvOS y watchOS

**Unity**: es un motor de videojuego multiplataforma creado por Unity Technologies. Unity está disponible como plataforma de desarrollo para Microsoft Windows, Mac OS, Linux. La plataforma de desarrollo tiene soporte de compilación con diferentes tipos de plataformas entre ellas móviles.

e) Se han identificado configuraciones que clasifican los dispositivos móviles en base a sus características.

Se podrían clasificar en dos grandes grupos, aunque seguramente se podrían clasificar de muchas otras maneras. Estos dos grupos serían los siguientes:

* **Propósito general**: Incluye dispositivos como computadoras, tablets, relojes y teléfonos inteligentes. Se caracterizan por su capacidad para realizar una amplia gama de actividades y sintetizar funciones. Son el medio de trabajo para muchas personas, permitiendo realizar actividades que generan alta productividad.
  + Teléfonos inteligentes o smartphones. Podemos decir que son como teléfonos móviles convencionales pero construidos sobre una estructura informática móvil, que les proporciona mayor capacidad de almacenamiento y de procesamiento. Se asemejan bastante a un ordenador. Son usados a menudo tanto para el entretenimiento como herramientas de trabajo.
  + Tabletas. Son ordenadores portátiles personales integrados en una pantalla táctil, con acceso a Internet y capacidad para ejecutar aplicaciones instaladas sobre un sistema operativo. Su manejo es fácil e intuitivo.
  + Relojes inteligentes. Son relojes de pulsera que poseen funcionalidades como acceso a Internet, recibir llamadas o enviar mensajes.
  + Ordenadores portátiles. Son ordenadores completos que, debido a su tamaño reducido y compacto, permiten su transporte y gracias a su batería, su uso mientras no esté conectado el cargador a la red eléctrica.
* **Propósito de entretenimiento:** Este grupo incluye dispositivos diseñados para proporcionar diversión a las personas, como aquellos que permiten jugar videojuegos como propósito principal.
  + **Nintendo Switch:** es una consola de videojuegos desarrollada por Nintendo. Fue lanzada mundialmente el 3 de marzo de 2017.
  + **Steam Deck**: es un [PC de juegos](https://es.wikipedia.org/wiki/PC_de_juegos) portátil desarrollado por [Valve Corporación](https://es.wikipedia.org/wiki/Valve_Corporation). Lanzado el 25 de febrero de 2022, Steam Deck se puede usar como PC portátil o de sobremesa usando un monitor externo. El dispositivo está diseñada para soportar gran parte de la biblioteca de Steam utilizando [SteamOS](https://es.wikipedia.org/wiki/SteamOS), un sistema operativo que estriba en un kernel Debian[1](https://es.wikipedia.org/wiki/Steam_Deck#cite_note-1)​. Esto permite el soporte de un gran número de juegos de PC de [Windows](https://es.wikipedia.org/wiki/Windows) a través de la capa de compatibilidad [Proton](https://es.wikipedia.org/wiki/Proton_(software)) utilizando [Linux](https://es.wikipedia.org/wiki/Linux). El sistema es una plataforma abierta, permitiendo a los usuarios instalar otros juegos compatibles con el dispositivo o incluso otros sistemas operativos.

g) Se ha analizado la estructura de aplicaciones existentes para dispositivos móviles identificando las clases utilizadas.

Analizar las clases que tenemos en el pong

h) Se han realizado modificaciones sobre aplicaciones existentes.

Revisar la documentación y entregada del pong y comprobarla.

i) Se han utilizado emuladores para comprobar el funcionamiento de las aplicaciones.

Explicar emuladores mercado

Switch yuzu.

BlueStack.

Emulador móvil unity (hacer foto)

**RA2**

a) Se ha generado la estructura de clases necesaria para la aplicación.

Orden lógico de los Scripts

Sale del juego

b) Se han analizado y utilizado las clases que modelan ventanas, menús, alertas y controles para el desarrollo de aplicaciones gráficas sencillas.

Explicar el canvas menú principal

c) Se han utilizado las clases necesarias para la conexión y comunicación con dispositivos inalámbricos.

investigar

d) Se han utilizado las clases necesarias para el intercambio de mensajes de texto y multimedia.

investigar

e) Se han utilizado las clases necesarias para establecer conexiones y comunicaciones HTTP y HTTPS.

investigar

f) Se han utilizado las clases necesarias para establecer conexiones con almacenes de datos garantizando la persistencia.

investigar

Clase para realizar conexión a BBDD

g) Se han realizado pruebas de interacción usuario-aplicación para optimizar las aplicaciones desarrolladas a partir de emuladores.

Hacer Beta Test!

h) Se han empaquetado y desplegado las aplicaciones desarrolladas en dispositivos móviles reales.

Se puede hacer el APK y probar.

i) Se han documentado los procesos necesarios para el desarrollo de las aplicaciones.

Fotos de captura de GDD, foto de comentario de código….