Índice

[Conceptos 2](#__RefHeading___Toc110_1439892024)

[Arquitecturas de Sofwares 7](#__RefHeading___Toc112_1439892024)

[Arquitectura Desktop 7](#__RefHeading___Toc116_1439892024)

[Desktop Windows 7](#__RefHeading___Toc118_1439892024)

[Desktop Linux 8](#__RefHeading___Toc438_1604927436)

[Desktop Andriod o Mobile Andriod 8](#__RefHeading___Toc457_3679562450)

[Arquitectura Desktop Hosting 8](#__RefHeading___Toc120_1439892024)

Índice de ilustraciones

[Ilustración 1: Red Informática 2](#Ilustraci%C3%B3n!0|sequence)

[Ilustración 2: Servidor 2](#Ilustraci%C3%B3n!1|sequence)

[Ilustración 3: Servidor Web 3](#Ilustraci%C3%B3n!2|sequence)

[Ilustración 4: Arquitectura de Software 3](#Ilustraci%C3%B3n!3|sequence)

[Ilustración 5: Aplicación de Escritorio 4](#Ilustraci%C3%B3n!4|sequence)

[Ilustración 6: Aplicación Multiplataforma 4](#Ilustraci%C3%B3n!5|sequence)

[Ilustración 7: Python 4](#Ilustraci%C3%B3n!6|sequence)

[Ilustración 8: Java 5](#Ilustraci%C3%B3n!7|sequence)

[Ilustración 9: C# 5](#Ilustraci%C3%B3n!8|sequence)

[Ilustración 10: Software Libre 5](#Ilustraci%C3%B3n!9|sequence)

[Ilustración 11: Linux 6](#Ilustraci%C3%B3n!10|sequence)

[Ilustración 12: Microsoft Windows 6](#Ilustraci%C3%B3n!11|sequence)

[Ilustración 13: Macintosh Apple Inc. 6](#Ilustraci%C3%B3n!12|sequence)

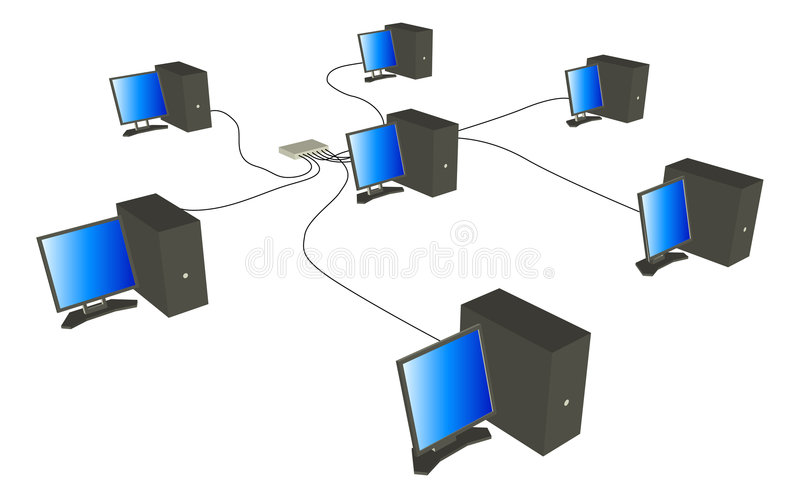
[Ilustración 14: Android 7](#Ilustraci%C3%B3n!13|sequence)

[Ilustración 15: Desktop Windows 8](#Ilustraci%C3%B3n!14|sequence)

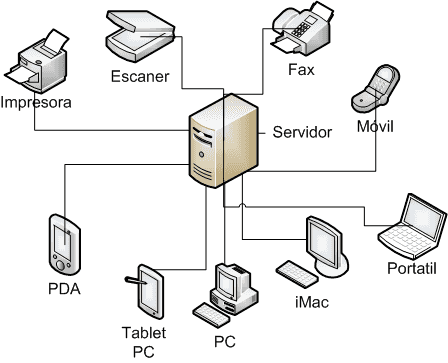
[Ilustración 16: Desktop Linux 8](#Ilustraci%C3%B3n!15|sequence)

1. Conceptos

**Red informática:** es un conjunto de equipos nodos y softwares conectados entre si por medio de dispositivos físicos que envían y reciben impulsos eléctricos, ondas electromagnéticas o cualquier otro medio para el transporte de datos, con la finalidad de compartir información, recursos y ofrecer servicios.

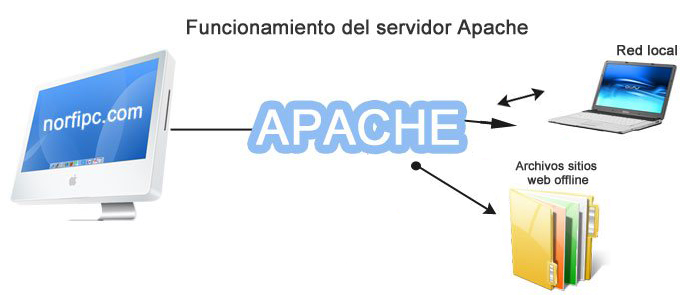
  
Ilustración 1: Red Informática

**Servidor:** Ordenador que es compartido, en una red informática, por múltiples usuarios.

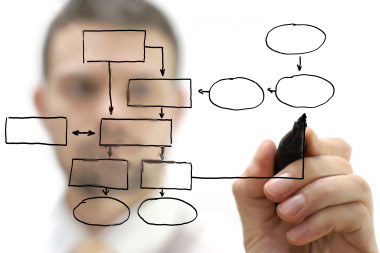
  
Ilustración 2: Servidor

**Servidor Web:** Una programa informático que procesa una aplicación del lado del servidor, realizando conexiones con el cliente y generando o cediendo una respuesta en cualquier lenguaje o aplicación del lado del cliente. El código recibido por el cliente es renderizado por un navegador web. El termino también se emplea para referirse al ordenador que realiza la acción de servir un sitio web.

**Arquitecturas de softwares:** Formas y guías generales, con base a las cuales se puedan resolver los problemas. Indican la estructura, funcionamiento e interacción entre las partes del software.

  
Ilustración 3: Servidor Web

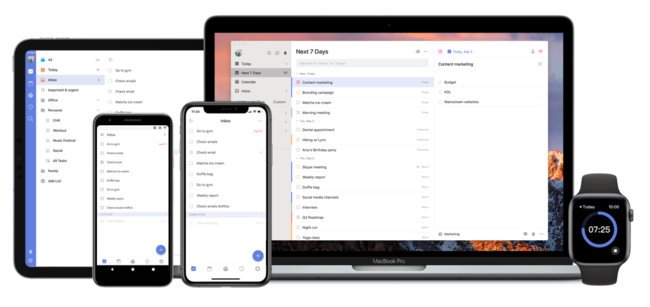
**Aplicaciones de escritorio:** es aquella que se encuentra instala en el ordenador o sistema de almacenamiento (USB) y podemos ejecutarlo sin Internet en nuestro sistema operativo, al contrario que las aplicaciones en al nube que se encuentran en otro ordenador (servidor) al que acedemos a través de la red o Internet a su software.

  
Ilustración 4: Arquitectura de Software

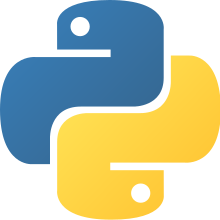
**Aplicación multiplataforma:** una aplicación multiplataforma puede ejecutarse sin problema alguno, tanto en Microsoft Windows en la arquitectura x86, como en Linux en la arquitectura x86 y Mac OS X, ya sea en el Power PC o sistemas Apple Macintosh basados en x86. En general, una aplicación multiplataforma se puede ejecutar tanto en todas las plataformas existentes o como mínimo en dos plataformas.

  
Ilustración 5: Aplicación de Escritorio

**Python:** Es un lenguaje de alto nivel de programación interpretado cuya filosofía hace hincapié en al legibilidad de su código, se utiliza para desarrollar aplicaciones de todo tipo, incluido web y de escritorio. Se trata de un lenguaje de programación multiparadigma. Es un lenguaje interpretado, dinámico y multiplataforma.

  
Ilustración 6: Aplicación Multiplataforma

**Java:** es un lenguaje de programación y una plataforma informática. Java es rápido, seguro y fiable. Desde ordenadores portátiles hasta centros de datos, desde consolas para juegos hasta computadoras avanzadas, desde teléfonos móviles hasta Internet, Java está en todas partes. Si es ejecutado en un plataforma no tiene que ser recompilado para correr en otra. Java es a partir de 2012, uno de los lenguajes de programación mas populares en uso, particularmente para aplicaciones de cliente servidor de web. Con unos diez millones de usuarios reportados.

  
Ilustración 7: Python

**C#:** Es un lenguaje de programación multiparadigma desarrollado y estandarizado por la empresa Microsoft como parte de su plataforma .NET, que después fue aprobado como estándar. Fue diseñado para la infraestructura de lenguaje común.

  
Ilustración 8: Java

**Software libre:** el código fuente puede ser estudiado, modificado y utilizado libremente con cualquier finalidad (gratis) y redistribuido con cambios o mejoras sobre ellas.

  
Ilustración 9: C#

**Linux:** denominación técnica y generalizada que reciben una serie de sistemas operativos de tipo Unix, que también suelen ser de código abierto, multiplataforma, multiusuario y multitarea. Estos sistemas operativos están formado mediante la combinación de varios proyectos, entre los que se destaca el entorno GNU y una fundación cuyo propósito es difundir el software libre, así como también el núcleo de sistema operativo conocido como Linux.

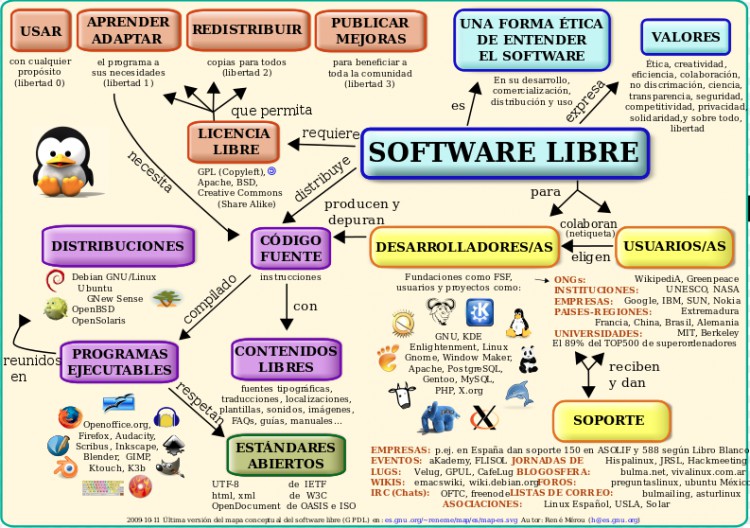
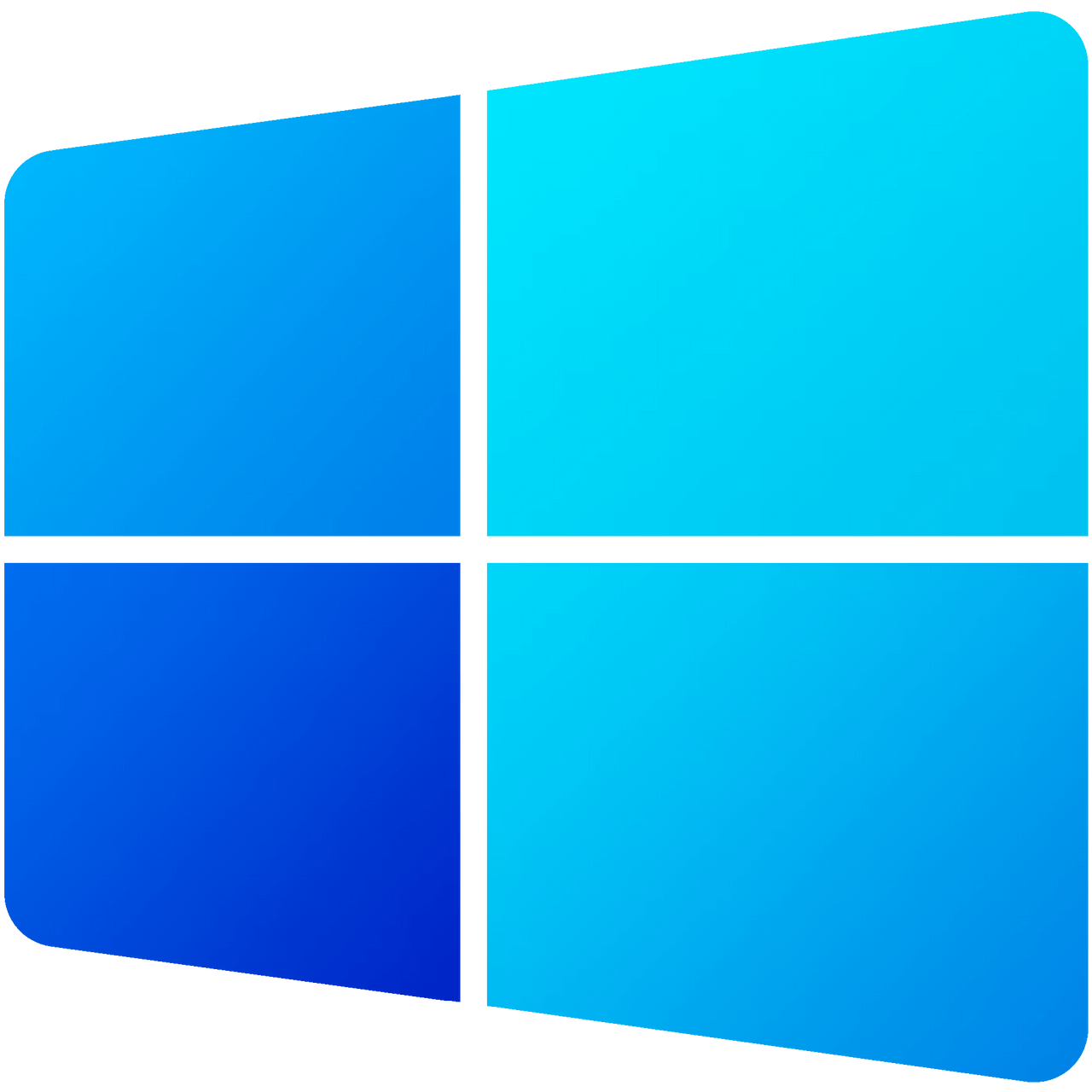
  
Ilustración 10: Software Libre

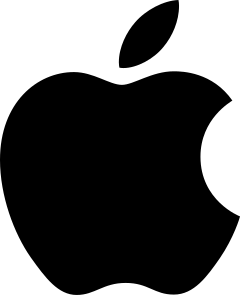
  
Ilustración 11: Linux

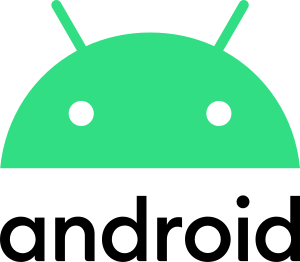
**Microsoft Windows:** nombre de una familia de distribuciones de software para PC, servidores, sistemas empotrados y antiguamente teléfonos inteligentes desarrollados y vendidos por Microsoft y disponibles para múltiples arquitecturas como x86, x86-64 y ARM. No es software libre por lo tanto es de pago y su código no es abierto.

  
Ilustración 12: Microsoft Windows

**Macintosh:** abreviado como MAC, es la linea de ordenadores personales diseñada, desarrollada y comercializada por Apple Inc. No es software libre y es bien caro.

Android: sistema operativo móvil basado en el núcleo Linux y otros de código abierto. Fue diseñado para dispositivos móviles con pantalla táctil, como teléfonos inteligentes, tabletas , relojes inteligentes Wear OS, automóviles con otros sistemas a través de Android Auto, al igual los automóviles con el sistema Android Automotive y televisores Android TV.

  
Ilustración 13: Macintosh Apple Inc.

  
Ilustración 14: Android

1. Arquitecturas de Sofwares
   1. Arquitectura Desktop

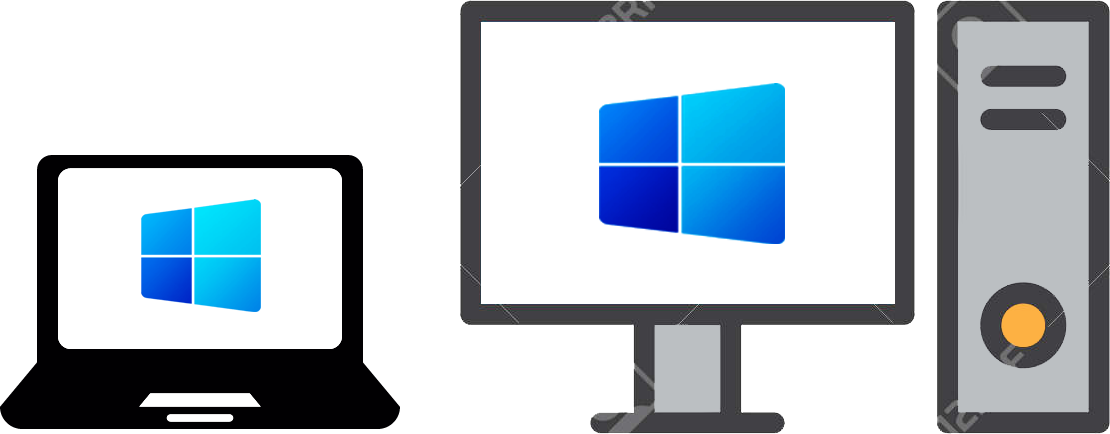
Esta basada en aplicaciones de escritorio. Se utilizan lenguajes de todo tipo para materializare siendo Python, Java y C# los más comunes. Su estructura se basa en aplicaciones que estén programadas para trabajar dependiendo solo de la plataforma y no de ningún servicio de red u/o Internet; o sea aplicaciones que solo necesitan del sistema operativo Microsoft Windows, Linux o Mac. La aplicación sera programada y se usara solo en el dispositivo que se instalara ya sea PC o móvil.

* + 1. Desktop Windows

Para lograr esta arquitectura debemos usar lenguajes como Java, Python, C#. Es la más común y la que todo programador o usuario estándar esta acostumbrado ya que casi todos saben utilizar. Se debe usar en el sistema operativo Microsoft Windows que es un software privativo por el cual debemos pagar licencias. Microsoft contiene una cartera de diversos sistemas operativos para el uso de despliegues de aplicaciones, por ejemplo:

* Microsoft Windows 11
* Microsoft Windows 10
* Microsoft Windows 8.1
* Microsoft Windows Server 2019

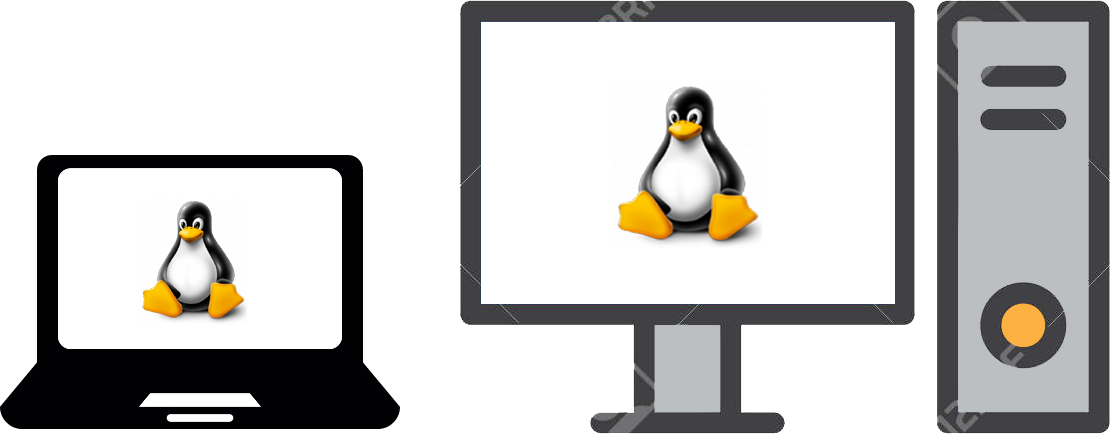
Desktop Linux

  
Ilustración 15: Desktop Windows

Para lograr esta arquitectura debemos usar lenguajes como Java, Python, C# pero este ultimo presenta algunos inconvenientes para esta arquitectura ya que viene preparado para Windows. Es uno de los menos comunes, los programadores deben estar acostumbrados ya que es el sistema operativo mas común para desarrollar; pero es el menos usado para usuarios estándar como PC de trabajo o de ocio. Existen diversas distribuciones de Linux:

* Debian
* Ubuntu
* Fedora
* OpenSUSE
* Arch Linux
* Manjaro
* Otras muchas más.

Desktop Andriod o Mobile Andriod

  
Ilustración 16: Desktop Linux

Para lograr esta arquitectura debemos usar lenguajes como Java y C#. Es una de las arquitecturas mas usadas ya que es libre y es el más distribuido en móviles y tablets. El problema de esta arquitectura es el constante actualización del sistema Android esto a su vez imposibilita que la aplicación perdure en el tiempo ya que su instalación funcional no esta garantizada para versiones futuras y se deba actualizar el programa almenos anualmente. Esta arquitectura ademas tiene un problema: mi equipo y yo tenemos 0 experiencia en su desarrollo, no significa que no se pueda lograr sino que se demora mas de lo normal en lanzar la aplicación.

Ilustración 17: Desktop Android o Mobile Android

* 1. Arquitectura Desktop Hosting
  2. Esta arquitectura compone todas las arquitecturas antes mencionadas menos la Desktop Android o Mobile Android. La síntesis de esta arquitectura es que el software funcione en el dispositivo sin importar el sistema operativo donde pueda realizar todas sus funciones y a su vez tenga la opción de servir por vía web la misma aplicación con la totalidad o parcialidad de sus funciones.