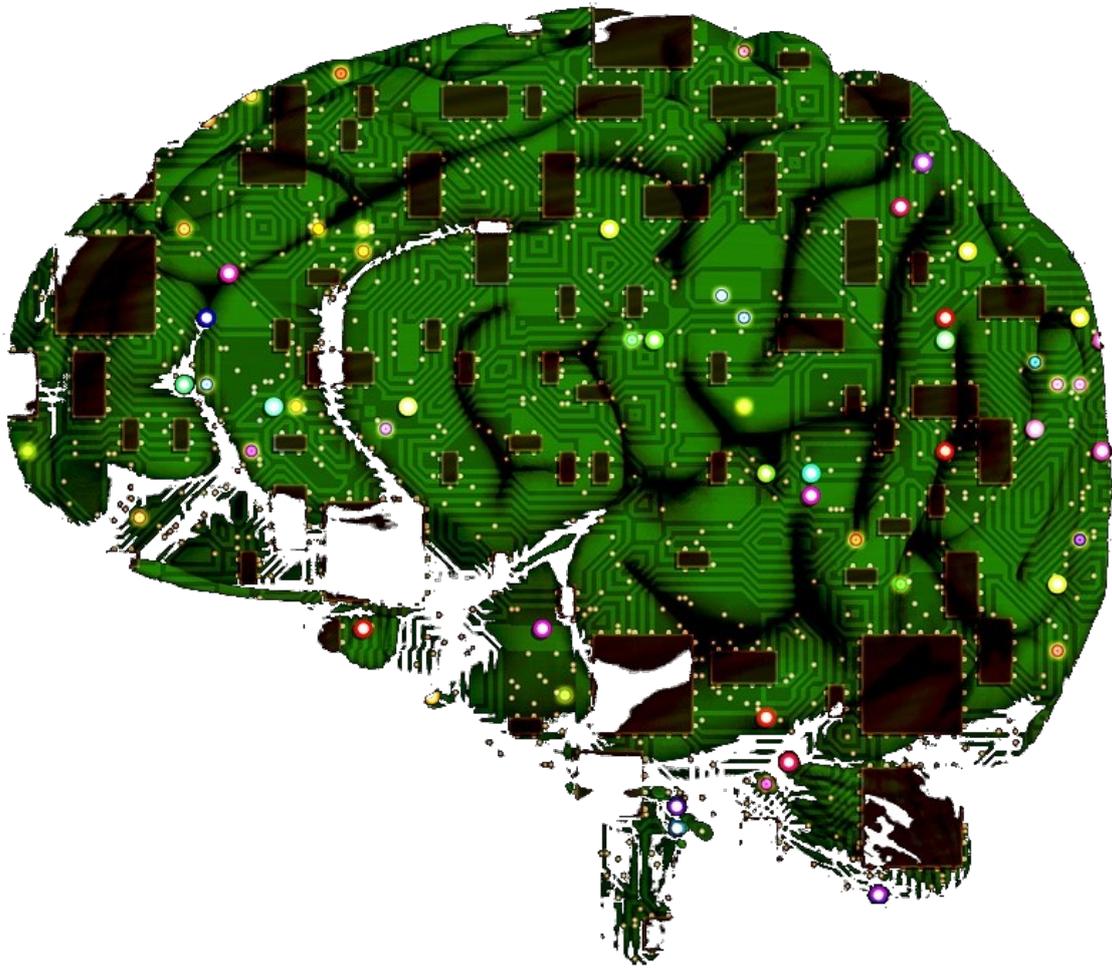


Material didáctico

Software

Magister Ricardo P. Salvador



2017

1. Introducción

Software es el conjunto de instrucciones y demás recursos simbólicos necesarios para hacer funcionar una computadora, un teléfono móvil, una tablet, un reloj inteligente, y todo dispositivo con posibilidades de ejecutar instrucciones automáticamente. Mediante software se implementan los modelos de inteligencia artificial que realizan cada vez más tareas antes sólo exclusivas para seres humanos por su complejidad, por ejemplo, reconocer una cara (hace unos años, nuestros celulares nos *reconocen* mediante un *algoritmo de reconocimiento facial*). Es un conjunto bastante amplio como lo que podemos hacer con una computadora; dicho de otra manera, todo lo que podemos hacer con computadoras requiere de software específico para ello. Hay software para jugar, dibujar, sacar fotos, escuchar y grabar música, comunicarnos con otros, escribir, hacer cálculos, software para que funcione otro software, y software ¡para construir nuevo software!



Ilustración 2: <http://www.stockxpert.com/browse.phtml?f=view&id=94398>

Índice

1. Introducción.....	2
2. Qué es el <i>software</i>	4
3. ¿Dónde se "encuentra" el software?.....	5
4. ¿Cómo se produce el software?.....	5
5. El software omnipresente y la importancia de la programación.....	6
6. La ingeniería de software.....	7
7. Clases de software.....	7
1. Clasificación según la función que realiza:.....	7
2. Clases según la licencia de uso.....	9
3. Clases según la influencia del usuario en su elaboración.....	10
8. El software que usamos.....	10
9. Sistemas operativos.....	11
1. Android.....	12
2. MS Windows.....	12
3. El universo GNU/Linux.....	13
4. Apple iOS y Mac OS.....	14
5. Control total: <i>root</i> o <i>jailbreak</i>	15
10. Comparativa de instalaciones de Linux, Windows y MacOS.....	15
11. Comparativa entre Android, Windows Mobile y iOS.....	16
12. Comparativa de sistemas operativos en servidores.....	16
13. Piratería de software: cuestiones legales.....	17
14. Vulnerabilidad del software.....	19
15. Conclusión.....	20
16. Bibliografía.....	20

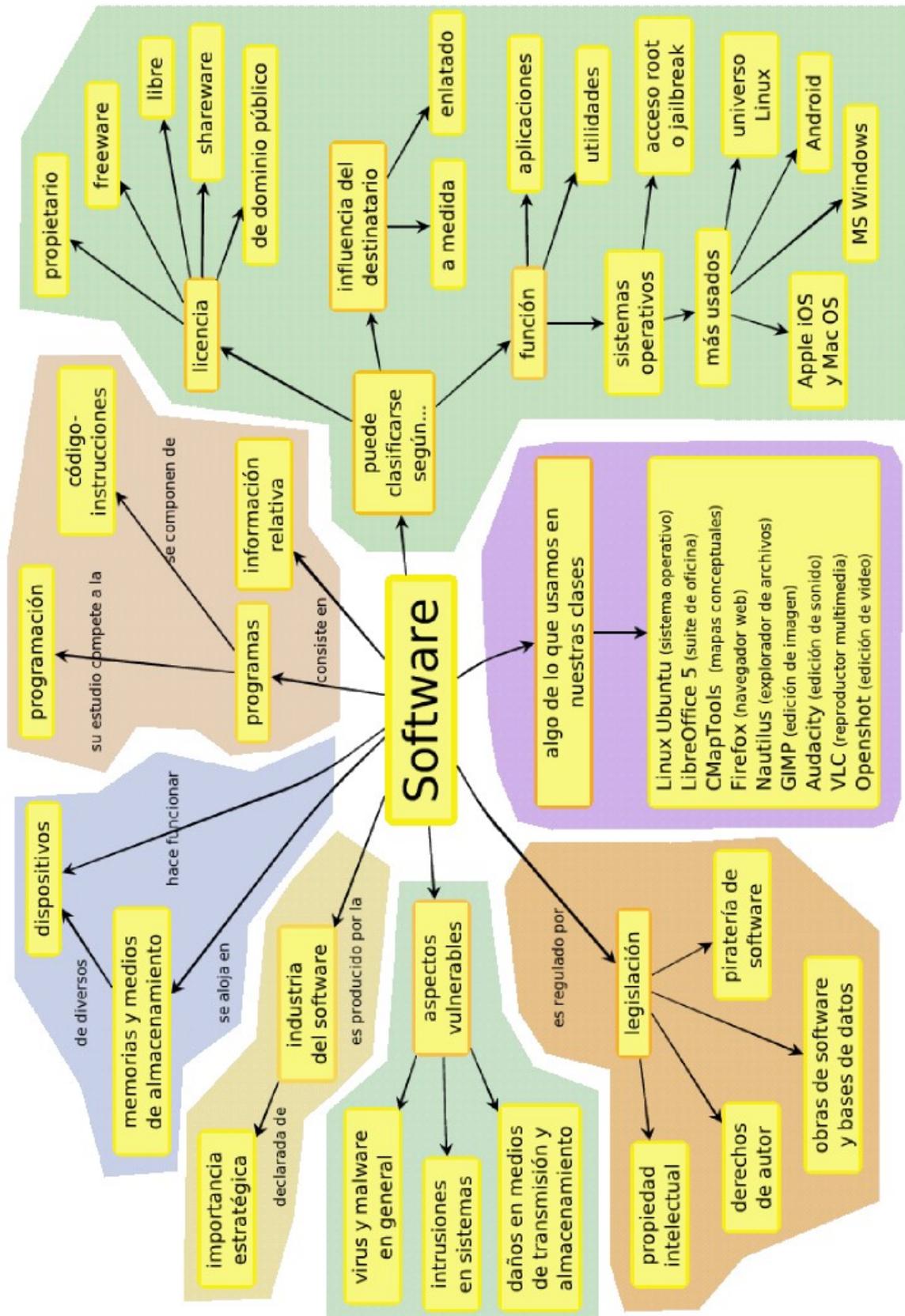


Ilustración 3: Representación gráfica de las principales ideas de este apunte

2. Qué es el software

Entré en el *ciber*, apurado por mandar ese mail, y me encontré con todas las computadoras ocupadas. -"Ya se desocupa la 14"- dice el encargado. Y así mientras espero, atento a "la 14", me distraigo con la escena del ciber. Todas las computadoras son iguales, y tanto como en su aspecto, seguramente son iguales por "dentro": su velocidad, la cantidad de memoria, sus componentes, etc... Todo bastante uniforme, salvo porque un usuario está jugando una carrera de autos en 3D, otro parece leer un diario on-line, el otro escribe lentamente, como pensando cada letra... y cada uno de los demás hace algo distinto, como burlándose del igual y repetido *hardware* que hay ahí.



Cosas iguales, cosas distintas. Todas esas computadoras iguales en apariencia, y cada usuario haciendo, con cada una, una tarea distinta. Y más, cada tanto, el mismo usuario cambia de tarea y pasa de un juego a revisar su correo electrónico, a revisar un fotolog, y más tarde, ve las últimas noticias.

¿Cómo **una misma computadora** se puede utilizar para tantas cosas distintas? La respuesta se puede resumir en estas palabras: **es una máquina programable**.

A semejanza de un ascensor, que, tras ser indicados los pisos a los que se desea ir, se dirige a uno, al siguiente, al siguiente, y así mientras sigue recibiendo "instrucciones", **una computadora realiza una tarea sólo si se le proporcionan las instrucciones necesarias para ello**; dicho en otras palabras, una computadora funcionará de acuerdo al *programa de instrucciones* que se le proporcione; y en otras palabras, lo que podemos hacer con una computadora depende de que tengamos el *programa* específico para esa tarea.

Las instrucciones que hacen funcionar una computadora son increíblemente más complejas, extensas y variadas que las de un ascensor, al punto que hay profesionales que se especializan sólo en uno u otro tipo de esas instrucciones.

Así, como una empresa que agrupa sus personal en diversas funciones, las instrucciones destinadas a realizar determinada tarea se agrupan en "paquetes" que pueden archivar y que se denominan **programas**.

Además, **una computadora** tiene una característica que la distingue de otras máquinas: **ejecuta las instrucciones de los programas en forma automática**, es decir, sin necesidad de que intervengamos. Esto es tan distintivo de las computadoras que, cuando por ejemplo, un

automóvil realiza ciertas tareas automáticamente, al dispositivo se lo denomina "computadora".

Y con programas de computadora podemos hacer tareas muy complejas: jugar juegos de estrategia, diseñar proyectos de arquitectura, conectar computadoras formando grandes redes... y la lista es casi interminable. Y cada programa tiene *procedimientos* propios para usarlo, *necesidades de hardware* (velocidad, memoria, dispositivos de hardware específicos, capacidad de disco rígido...) y requiere de otros programas previamente instalados, en principio, determinado *sistema operativo* y muchas veces otros auxiliares. Estas características de un programa se detallan en *manuales* sea impresos o *digitales*, que acompañan a cada programa.

Al universo formado por programas de computadora y los procedimientos, información, manuales y documentación relacionada se lo denomina **software**. Esta es una definición rigurosa de **software**, palabra que también suele utilizarse como sinónimo de *programa*.

3. ¿Dónde se "encuentra" el software?

Como toda información digital el software se almacena en (archiconocidos) medios tales como: disquetes, discos rígidos, CD-ROM, DVD-ROM, *pendrives*, la misma *nube* y cualquier dispositivo que soporte un sistema de archivos.

Ya que podemos bajar software de Internet, ¿es Internet un "lugar", un medio de almacenamiento? En rigor, el software que se encuentra *en Internet* en última instancia se encuentra almacenado en los discos rígidos de algún servidor de esa (gran) red, que en realidad es el medio de comunicación por el que podemos "descargarnos" o enviar información.



Ilustración 4:
https://upload.wikimedia.org/wikipedia/commons/d/d4/America_noviter_delineata.jpg

4. ¿Cómo se produce el software?

Si bien el *software* comprende programas, procedimientos, datos, documentación, al pensar en la producción de software se piensa en los *programas*, ya que éstos determinan a los demás componentes.

Los programas se componen de instrucciones, y mientras que en los primeros computadores las instrucciones se introducían una a una y se ejecutaban directamente, actualmente las instrucciones se escriben, se guardan en archivos, y luego pueden ejecutarse *automáticamente* a partir del archivo que las



Ilustración 5:
<http://www.stockxpert.com/browse.shtml?f=view&id=271319>

contiene.

Las instrucciones para crear programas, o *programar*, forman conjuntos llamados *lenguajes de programación*, y existen *lenguajes* para escribir software de propósitos más o menos específicos, por ejemplo: *Lenguaje C*, *Visual BASIC*, *Macromedia Flash*, *Assembler*, *Java*, *PHP...* y la lista es muy larga.

Cada *lenguaje de programación* tiene sus reglas de trabajo, y a la hora de hacer un *programa*, se parte del *problema* que hay que resolver y se plantea el programa según los recursos del lenguaje a utilizar.

5. El software omnipresente y la importancia de la programación

El software está en todas partes. Es fácil imaginar la relación directa de software y, por ejemplo, Linux, Android, Microsoft Windows, Amazon.com, Whatsapp, Twitter, Google, LibreOffice.org, Facebook, Java y Snapchat. Y que hace funcionar computadoras, tablets, teléfonos móviles. También se encuentra en los automóviles (GPS, telefonía, control de tracción, inyección de combustible...), en una pierna o en un brazo biónicos, en exoesqueletos, en drones, en un robot de cirugía, en un chip para fotografía de arterias y venas, en dispositivos de agricultura de precisión, y en las 10.000 resmas A4 de instrucciones que controlan un Airbus 100 M.



Ilustración 6:
<http://www.stockxpert.com/browse.phtml?f=view&id=283085>

La ubicuidad del software es tal que aprender a programar como herramienta intelectual y formar desarrolladores de software es necesario para entender, gestionar y crecer en el mundo tecnológico, para producir productos de alto valor agregado, para reducir la dependencia tecnológica y fortalecer la soberanía. En nuestro país el sector del software es uno de los que históricamente ve insatisfecha la demanda de profesionales, motivo por los cuales la Presidencia de la Nación y la fundación Sadosky impulsan iniciativas de promoción de la programación y el pensamiento computacional, tales como "Dale aceptar" (<http://www.daleaceptar.gob.ar>) y Program.Ar (program.ar). Y más recientemente, en 2016, se lanzó el "Plan 111 Mil", un plan nacional que se propone formar en los próximos 4 años a 100.000 programadores, 10.000 profesionales y 1.000 emprendedores para cubrir la demanda laboral de las Industrias Basadas en el Conocimiento.

Podés leer más en:

- Mendarozqueta, Álvaro (2017), "¿Por qué hay que aprender a programar?", <https://es.slideshare.net/AlvaroRuizdeMendaroz/por-qu-hay-que-aprender-a-programar-una-actualizacion>

6. La ingeniería de software

Existe software con diversa complejidad. Por un lado podemos encontrar un juego como el famoso "Buscaminas", que puede ser ideado y programado por una sola persona. Por otra parte, otros software son mucho más complejos, como los sistemas operativos, los paquetes de software de oficina que incluyen varias aplicaciones complejas, o los sistemas hechos a medida para grandes instituciones, este tipo de software requiere equipos de profesionales de diferentes campos trabajando sobre ese mismo proyecto (durante años quizás) para conocer las necesidades de los usuarios, establecer las características que tendrá el producto, diseñar el software, implementarlo, ponerlo a prueba y mantenerlo funcionando a lo largo del tiempo. Además el producto se pone a prueba, se reciben opiniones de los usuarios, se modifica; cada tanto, se ofrecen nuevas *versiones*, que incorporan mejoras, dejan atrás errores, se adaptan a los nuevos sistemas operativos. A medida que la producción de software se hizo más compleja, se la consideró una nueva ingeniería: la *ingeniería de software*, una especialización dentro de las Ciencias de la Computación.



Ilustración 7:
<http://www.stockxpert.com/browse.phtml?f=view&id=174276>

7. Clases de software

Pensando en las diversas actividades que hacemos con una computadora encontraremos que hay diversas clases de software; veremos a continuación dos clasificaciones: *a.* según la función que realiza, *b.* según el modo de distribución.



Ilustración 8:
<http://www.stockxpert.com/browse.phtml?f=view&id=298311>

1. Clasificación según la función que realiza:

a. Aplicaciones: juegos, tareas de oficina, navegar en Internet, dibujar, etc., son programas con los que el usuario realiza una tarea específica, relativa a un objetivo del propio usuario. Son tareas centradas en el interés del usuario. Tan diverso como lo que puede hacer un usuario con su computador, es el conjunto de aplicaciones; así se pueden clasificar, por ejemplo, en aplicaciones de automatización industrial, para automatizar tareas de oficina (ofimática), software educativo, software médico, aplicaciones para gestionar grandes bases

de datos, juegos, etc.; la lista puede ser muy larga...

b. Utilidades: este software se utiliza para realizar tareas de mantenimiento y mejorar el rendimiento del mismo computador, tanto de su hardware como de su software; este software está centrado en el computador.

c. Lenguajes de programación: es software que se utiliza para crear nuevo software; consiste en un repertorio de instrucciones y reglas de uso con los que se escriben los nuevos programas, código que luego se traducirá al código de máquina (más parecido a los "ceros" y "unos"). Utilizar "ceros" y "unos" para escribir instrucciones fue una tarea habitual en computadores más simples que los actuales: paradójicamente no eran computadores personales sino "grandes computadores" usados en grandes organizaciones, pero su capacidad de cálculo y almacenamiento era muy, pero muy limitada comparada con los actuales computadores de escritorio o personales. Ejemplos de lenguajes de programación son *C, Pascal, Basic, Visual Basic, Java, Perl, Assembler, Logo, Prolog*, y esta lista también es bastante larga.

Las imágenes que siguen muestran el código de tres lenguajes de programación: ¿adivinan cuáles están destinados a los niños y jóvenes?

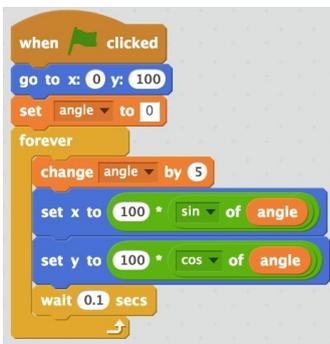


Ilustración 10: Scratch
<https://www.flickr.com/photos/wfryer/9461183065>

```
def add5(x):
    return x+5

def dotwrite(ast):
    nodename = getNodeName()
    label=symbol.sym_name.get(int(ast[0]),ast[0])
    print ' %s [%label="%s" % (nodename, label),
    if isinstance(ast[1], str):
        if ast[1].strip():
            print '=' % ast[1]
        else:
            print ''
    else:
        print ''
        children = []
        for n, childrenumerate(ast[1:]):
            children.append(dotwrite(child))
        print , ' %s -> (' % nodename
        for n, namechildren
            print '%s' % name,
```

Ilustración 11: Python
<https://commons.wikimedia.org/wiki/File:Python.png>



Ilustración 9: Squeak-Etoys
 R.P.Salvador

Un ejemplo de programa e instrucciones

En los inicios de la programación las instrucciones se daban al computador mediante perforaciones en tarjetas de cartón o en cintas de papel y más tarde se escribían los "bits" directamente en la memoria RAM. Pero desde los años 50, si bien hace unos años es posible crear programas usando más el ratón que el teclado (por ejemplo, *Visual Basic*), la mayor parte de los *lenguajes*, desde los inicios de la programación hasta hoy, en algún momento requieren de instrucciones escritas parecidas a las que muestra la siguiente tabla:

1.	comenzar	indica el inicio del programa
2.	pantalla.borrar	borra la pantalla
3.	círculo.x ← 200	sitúa al objeto "círculo" en la posición horizontal 200
4.	círculo.y ← 100	establece su posición vertical en 100
5.	círculo.radio ← 70	establece su radio en 50 pixeles
6.	círculo.color ← amarillo	colorea al círculo en color <i>amarillo</i>
7.	terminar	indica que finaliza el programa

d. Sistemas operativos: Linux, Windows, D.O.S., Unix, son programas que todos los usuarios tienen pero ninguno resuelve muchos de sus problemas usando sólo estos software. Entonces, ¿por qué todos tenemos uno de estos software en nuestros equipos? Este tipo de programa permite que el usuario ingrese instrucciones e información al computador, controla el funcionamiento del hardware, y gestiona la ejecución de otros softwares (y de sí mismo). Por cumplir estas funciones necesarias para poder utilizar computador se lo denomina "*software de base*". Más adelante se desarrollan detalle sobre los sistemas operativos más usados.

2. Clases según la licencia de uso

a. Freeware: así se denomina al software gratuito. Es común que el *freeware* incluya una *licencia de uso* que permite su copia, informando a su autor, no permite su modificación, y muchas veces limita su uso, por ejemplo, a usuarios particulares. En la actualidad se puede "bajar" de Internet una gran cantidad de *freeware*, de muy buena calidad. La ventaja de no tener que pagar por el *freeware* tiene como contrapartida, algunas veces, la falta de soporte técnico por parte del autor, o la discontinuidad del mismo a lo largo del tiempo.

b. Software privativo o no libre: es el software cuya licencia de uso está sujeta a las leyes de propiedad intelectual y por lo tanto sólo permite su uso y aún en forma restringida, reservando el resto de los derechos para el propietario del copyright o derechos de autor.

c. Software libre: es el software cuya licencia de uso garantiza la libertad de *a.* usarlo con cualquier propósito y ámbito, *b.* analizarlo y modificarlo, *c.* redistribuirlo (copiarlo) y *d.* mejorarlo y publicar las mejoras.

Vale destacar que dentro del *software libre* hay sistemas operativos completos y numerosas aplicaciones.

Puede ser llamativo que se hagan grandes esfuerzos para ceder los *derechos* gratuitamente, pero es una tendencia que ha ganado fuerza en los últimos años frente al *software privativo*, y es fruto de *a.* una concepción sobre el software opuesta a la del *software privativo*, *b.* un tipo

de trabajo especialmente favorecido por Internet en el que participan una gran comunidad de programadores.

El ejemplo paradigmático de este tipo de software es el sistema operativo Linux, creado por el finlandés Linus Torvald en un proyecto abierto a la colaboración de una gran comunidad virtual de programadores; Linux se destaca por su estabilidad, y seguridad, utilizándose por esa característica como servidor en redes de computadoras.

d. Software de dominio público: los derechos de uso y explotación de este tipo de software han sido donados a la humanidad o han expirado; por este tipo de software no es necesario solicitar ninguna licencia y se puede hacer uso de él con fines legales y consignando su autoría original.

e. Shareware: es un software cuyo autor permite probar sin costo durante un período limitado de tiempo, o con limitación en sus funciones; finalizado el período de prueba o si el usuario decide adquirir el software "completo", entonces se comunica con el autor para acordar su compra. Es el caso de muchos programas que se pueden bajar de Internet.

3. Clases según la influencia del usuario en su elaboración

a. Software hecho a medida: es el caso del software que se crea cuando un analista de sistemas es contratado para elaborar un programa destinado a una situación específica, para la cual probablemente no exista otro software similar. El profesional que hará el software se entrevista con quien lo encarga y "ajusta" y personaliza el diseño del software a esa situación concreta. Podría decirse que es un software "artesanal".

b. Software "enlatado": es software que se crea para propósitos generales, destinado a una gran cantidad de usuarios. Las compañías que lo producen tienen en cuenta las necesidades de grandes grupos de usuarios, en contrario a lo que sucede con el software a medida; esto hace que sea relativamente menos costoso y de muy buena calidad. En general se adquiere es una copia y se paga una licencia para el uso de dicha copia. Su producción y comercialización puede compararse a los métodos industriales de producción.

8. El software que usamos

¿Y qué software usamos en nuestras aulas?

Desde 2014 en las aulas del Departamento de Tecnologías de la Información y la Comunicación se utiliza una configuración de software basada totalmente en software libre. La siguiente tabla muestra una correspondencia entre aplicaciones de uso habitual para el sistema operativo Microsoft Windows, y para Linux Ubuntu.

Aplicaciones	Windows	Linux Ubuntu 16.04
Bases de datos	Microsoft (MS) Office Access, LibreOffice Base	LibreOffice Base
Compresión de archivos	7zip , winrar	7zip, UnRAR
Edición y conversión de audio	Audacity	Audacity, SoundConverter
Conversión de video	Any Video Converter	Transmagedon
Conversión multimedia a formatos libres		OGG Converter
Dibujo	MS Paint GIMP	Gpaint, Pinta, GIMP, LibreOffice Draw
Grabación de DVD-CD	XCD Burner	Brasero, K3b
Hipertexto	MS Office Word / MS Powerpoint, LibreOffice Writer/ LibreOffice Impress	LibreOffice Writer/ LibreOffice Impress
Hoja de cálculo	MS Office Excel, LibreOffice Calc	LibreOffice Calc
Mapas conceptuales	Freemind , CMaps Tools	Freemind, CMaps Tools
Navegador de archivos	Windows Explorer	Nautilus, PCManFm
Navegador de Internet	Firefox , Google Chrome , Chromium	Firefox, Google Chrome , Chromium
Presentaciones multimedia	MS Office Powerpoint, LibreOffice Impress	LibreOffice Impress
Procesamiento de textos	MS Office Word, LibreOffice Writer	LibreOffice Writer
Reproductor multimedia	Windows Media Player, Videolan VLC	Videolan VLC
Edición de video	Windows Movie Maker	Openshot

En el cuadro anterior se proporciona acceso a para descargar cada aplicación freeware o de licencia libre (la mayoría). En el caso de las aplicaciones para Linux Ubuntu (la columna de la derecha), salvo excepciones (CmapsTools, Google Chrome), las aplicaciones se instalan desde el repositorio de Ubuntu, y de la misma manera se procede con en el caso de cualquier otra distribución.

9. Sistemas operativos

Así como el software en general hace funcionar cada vez más aspectos de nuestra existencia, los sistemas operativos en particular dividen las aguas del universo de hardware en lo que respecta a

licencias de uso tipos de dispositivo y usuario, por lo que a continuación se exponen características de los más usados en la actualidad.

1. Android

Android es un sistema operativo desarrollado por Google de para dispositivos móviles, esto es, teléfonos celulares, tables, y posteriormente, como ya sabemos, en relojes, televisores, autos...

Es de código abierto, lo que permite que los fabricantes de cada dispositivo le haga modificaciones, sobre todo en la interface (lo que se llama *capa de personalización*), que a su vez los usuarios podemos cambiar también. Éstos son dos fuertes motivos por los que android es **el sistema operativo móvil más usado**. Los programas para Android, si, las *aplicaciones* o *apps* en la jerga, están escritos en el lenguaje Java, y se descargan de la tienda oficial de Google, aunque si es posible también instalar los archivos (que tienen la extensión *.APK*) directamente.



Ilustración 12: Logo de Android
https://www.shutterstock.com/image-photo/kiev-ukraine-may-26-2015-android-283385381?irgwc=1&utm_medium=Affiliate&utm_campaign=FreeFoto&utm_source=39571

Podés leer más en:

- XatakaAndroid.com, “¿Qué es Android?”, <https://www.xatakandroid.com/sistema-operativo/que-es-android>
- The Tech Terms Computer Dictionary, “Android”, <https://techterms.com/definition/android>
- Web oficial de Android, <https://www.android.com/>

2. MS Windows

Windows es una serie de sistemas operativos desarrollados por Microsoft y es el sistema operativo **más instalado en computadores personales** en los últimos 20 años. Es de licencia Se ofrecen versiones diseñadas para diversos perfiles de usuarios y actividades, por ejemplo para el hogar, estudiantes, profesionales, y administración de redes de computadoras. La última versión es la 10, en coincidencia con la versión para celulares, Windows 10 Mobile.



Ilustración 13: Logo de Windows 10
<https://www.shutterstock.com/image-photo/kiev-ukraine-may-17-2016-hand-578419504?src=pCSpzGukumvkS3CX77TMNQ-1-10>

Éste último es el sistema operativo de Microsoft para dispositivos móviles y tabletas. Lanzado en 2015, es el sucesor de Windows Phone 8.1.

Podés leer más en:

- The Tech Terms Computer Dictionary, “Windows”, <https://techterms.com/definition/windows>
- Wikipedia, la enclopedia libre, “Microsoft Windows”, https://es.wikipedia.org/wiki/Microsoft_Windows
- Wikipedia, la enclopedia libre, “Windows 10 mobile”, https://es.wikipedia.org/wiki/Windows_10_Mobile

3. El universo GNU/Linux

Cuando se pregunta por "Linux" suelen escucharse muchas respuestas,

- programas de computación gratuitos,
- algo desconocido,
- un mundo oscuro que sólo conocen ciertos expertos
- lo que usan los nerd
- el software que usan los estudiantes universitarios de computación, porque es gratis



Ilustración 14: GNU y Tux bebés, mascotas de GNU y Linux <https://www.gnu.org/graphics/babies/BabyGnuTux-Big.png>

Más allá del significado exacto, Linux es una palabra relacionada parcialmente a esas respuestas, porque tiene que ver con

- un universo de software con características propias respecto de su licencia, que establece cuatro características esenciales:
 - ser usado como se desee y con cualquier propósito (hasta aquí, nada especial)
 - ser estudiado en su funcionamiento y modificado con total libertad (esto implica acceder a su código fuente)
 - distribuir copias libremente
 - distribuir libremente a terceros las copias modificadas

Las anteriores son conocidas como las cuatro "*libertades del software libre*".

Linux fue anunciado en agosto de 1991 por su creador Linus Torvalds, estudiante finlandés de ciencias de la computación, como un proyecto de nuevo sistema operativo en el que estaba trabajando a modo de entretenimiento. Es relativamente reciente, y durante esos años no fue muy conocido masivamente ya que habiendo surgido en el ámbito académico se convirtió en el sistema operativo de la abrumadora mayoría de los servidores de Internet y supercomputadoras del mundo, incluyendo Google, Amazon, la NASA, la máquina de Dios y la Estación Espacial Internacional; más recientemente se le sumaron los teléfonos con Android (si, Android funciona gracias a Linux, y son algunos miles de millones de teléfonos). Las aulas de Informática de nuestra Escuela funcionan 100% con Linux desde 2014.

En rigor, la creación de Linus Torvalds, Linux, fue el núcleo (kernel, en inglés) para un sistema operativo: el kernel de un sistema operativo es el conjunto de programas que permite a las aplicaciones acceder al hardware, gestionar recursos en general (memoria, almacenamiento, archivos, comunicaciones).

El resto de los componentes para llegar a un sistema operativo (bibliotecas, compiladores, editores de texto, interfaces de usuario) completo fue proporcionado por miles de programadores de todo el mundo, que, en forma colaborativa gracias a Internet, y a que la Free Software Foundation ya estaba trabajando en componentes para un sistema operativo libre, que sumados al

núcleo Linux, dieron origen al sistema operativo GNU/Linux.

A partir de este sistema operativo, gracias a su licencia GPL (General Public Licence - Licencia Pública General), surgen numerosos desarrollos que si bien se basan en GNU/Linux, aportan y se diferencian con particularidades propias. Por ejemplo, las más conocidas (y punto de partida de un sinnúmero de otras distribuciones):

- Debian
- Fedora
- Ubuntu
- SUSE
- Arch Linux
- Puppy Linux (para equipo con muy pocos recursos)
- Raspbian (Debian para la Raspberry Pi)

Para terminar, hay que destacar que si bien los sistemas GNU/Linux tienen una filosofía y dinámica de desarrollo diferentes que los sistemas comerciales (siendo MS Windows los mayoritarios), los lenguajes de programación que pueden utilizarse son C, C++, Java, Python, Perl, PHP y Ruby entre otros. Y entre las aplicaciones no libres disponibles se encuentran, Adobe Reader, Adobe Flash así como drivers para numerosas marcas de hardware.

A 25 años de sus inicios, Linux y el software libre en general han llegado para quedarse y dar espacio para usar, difundir, estudiar y crear software libremente, sumado a un espíritu colaborativo y solidario.

Podés leer más en:

- Debian.org, “¿Qué es GNU/Linux?”, <https://www.debian.org/releases/stable/mips/ch01s02.html.es>
- Wikipedia, la enclopedia libre, “Linux”, <https://es.wikipedia.org/wiki/GNU/Linux>
- Diario La Nación, “Linux cumple hoy 25 años de vida”, <http://www.lanacion.com.ar/1931149-linux-cumple-hoy-25-anos-de-vida>
- “¿Qué es el software libre?”, Página oficial del sistema operativo GNU, <https://www.gnu.org/philosophy/free-sw.es.html>
- The Tech Terms Computer Dictionary, “Linux”, <https://techterms.com/definition/linux>

4. Apple iOS y Mac OS

Mac OS es el sistema operativo de las computadoras de la empresa Apple Inc. desde el modelo Macintosh (1984), y fue reescrito desde cero en 2001. Actualmente va por la versión 10.12 (septiembre de 2016). Este sistema operativo tiene licencia propietaria.

Derivado Mac OS de iOS es el sistema operativo móvil con que Apple hace funcionar sus iPhones, iPads y iPods. Igual que Mac OS, es de licencia propietaria.

Podés leer más en:

- The Tech Terms Computer Dictionary, “Mac OS”, <https://techterms.com/definition/mac-os>
- Wikipedia, la enclopedia libre, “Mac OS”,



Ilustración 15:
<https://www.shutterstock.com/image-vector/apple-icon-332834984?src=ibimL5ecIP2sHy3h13ZFdA-1-1>

<https://es.wikipedia.org/wiki/MacOS>

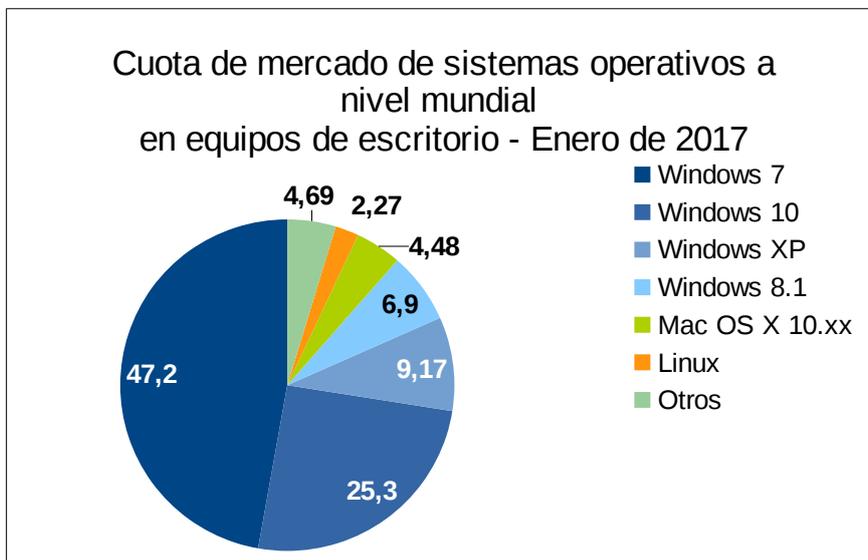
- The Tech Terms Computer Dictionary, "iOS", <https://techterms.com/definition/ios>

5. Control total: root o jailbreak

Los teléfonos celulares se venden con ciertas restricciones en su funcionalidad, que impiden que el usuario pueda obtener el rol de administrador o root, ejerciendo un control total del dispositivo. Esto ocurre en Android, iOS y Windows 10 Mobile, e impide al usuario instalar aplicaciones que no están en la tienda oficial o modificar el sistema en aspectos que no están disponibles en un equipo cuando se compra. El proceso de sortear esas modificaciones se denomina como jailbreak o root ("rooteo" en español), y... conlleva sus riesgos, por ejemplo, dejar inutilizable el teléfono ¡hasta encontrar la solución o alguien que sepa más!

10. Comparativa de instalaciones de Linux, Windows y MacOS

El siguiente gráfico muestra la cuota de mercado de los principales sistemas operativos a nivel mundial en computadores de escritorio a febrero de 2017, según el número de instalaciones. Windows 7 se sitúa por encima del 45% a pesar de que ya no se venden computadores que lo incluyan y Microsoft ya no lo actualiza, junto a las restantes versiones de Windows llegan al 88,57%, seguidos por Mac OS 10.12 y 10.11 con un 4,48%, el universo de sistemas Linux (2,27%) y el resto, 4,69%.



Podés leer más en:

- Statista – El portal de estadísticas, "Cuota de mercado de los principales sistemas operativos a nivel mundial a fecha de enero de 2017, según instalaciones", <https://es.statista.com/estadisticas/576870/cuota-de-mercado-mundial-de-los-sistemas-operativos>
- MuyComputer, "Windows 10 alcanza el 21% de cuota de mercado", <http://www.muycomputer.com/2016/08/02/windows-10-cuota-mercado/>
- Informática++, "Guía para elegir el sistema operativo de tu ordenador: ¿Windows, OS X o Linux?",

<http://informatica.blogs.uoc.edu/2016/03/08/guia-para-elegir-el-sistema-operativo-de-tu-ordenador-windows-os-x-o-linux/>

11. Comparativa entre Android, Windows Mobile y iOS

En el universo de los teléfonos móviles, la balanza está no inclinada, sino volcada hacia el lado de



Android: el sistema operativo del robotito verde hace funcionar al 86,2% de los dispositivos, iOS de Apple llega al 12,9%, Windows al 0,60% y el resto, al 0,30%.

Podés leer más en:

- ComputerHoy.com, "iOS y Android copan casi el 100% de la cuota de mercado", <http://computerhoy.com/noticias/moviles/ios-android-copan-casi-100-cuota-mercado-49794>

12. Comparativa de sistemas operativos en servidores

"Windows domina el escritorio, pero Linux gana el mundo". Esta frase, del editor jefe de ZDNet, Steve Ranger, refiere a que la abismal proporción de equipos de escritorio que usan alguna versión de Windows frente a la de Linux, se revierte cuando se examina el sector de servidores y supercomputadores. Según estadísticas del sitio W3Techs.com, la distribución es la que muestra el siguiente gráfico:



Podés leer más en:

- MuyComputer.com, "Windows domina el escritorio, pero Linux gana el mundo", <http://www.muycomputer.com/2017/02/20/linux-gana-el-mundo>
- W3Techs.com, "Usage of operating systems for websites", https://w3techs.com/technologies/overview/operating_system/all

13. Piratería de software: cuestiones legales

No pocos conflictos ha generado la circulación de música en formato MP3 por la Red, entre empresas discográficas y algunas empresas creadoras del software para compartir MP3.

Igualmente, el uso de copias ilegales de software ("software pirata") está penado por la ley.

Por eso una de las ventajas del software libre es que nos mantiene dentro de los límites de la legalidad.



Ilustración 16:
<http://www.stockxpert.com/browse.phtml?f=view&id=314444>

La legislación que legisla al software

En la República Argentina distintos aspectos del software están regulados por una sucesión de leyes y decretos nacionales entre 1933 y 2014:

- régimen legal de la propiedad intelectual
- qué se entiende por el término "obras de software"
- creación, el uso, y la difusión del software,
- declaración de su importancia estratégica para el desarrollo nacional

a. Ley 11.723 "Régimen Legal de la Propiedad Intelectual" (1933)

La ley 11.723 establece el Régimen de Propiedad Intelectual determinando

- qué es lo que se protege,
- quiénes son los titulares del derecho de autor
- y cuáles son los derechos que puede ejercer el titular de la obra, por ejemplo, a exigir que se indique su nombre o seudónimo en la misma y a exigir fidelidad en su texto, título, copias o reproducciones; por otro lado tiene derechos patrimoniales derivados de "la disposición, publicación, ejecución, representación, exposición, enajenación, traducción, adaptación, autorización de traducción y reproducción en cualquier forma de la misma" (Art.4)

b. El derecho de autor (Copyright), un derecho esencial

El derecho de autor o "Copyright" es una de las ramas que componen el derecho de propiedad intelectual. Consiste en el conjunto de facultades exclusivas que detentan los sujetos creadores de bienes intelectuales en los campos literario, artístico y científico, cualesquiera sean sus modos de expresión, mérito o destino.aaa

c. Decreto 165/94 – Protección para las obras de software y base de datos (1994)

Este decreto de Poder Ejecutivo Nacional especifica qué se entenderá por obras de software en el artículo 1° de la ley 11.723:

- los diseños del flujo lógico de los datos en un sistema de computación
- los códigos fuente y objeto
- documentación técnica relativa a explicar, entrenar, desarrollar, usar o mantener el software

d. Ley 25.036. Propiedad intelectual. Programas de computación fuente y objeto (1998)

Esta ley, sancionada en 1998, modifica la ley 11.723 (régimen de Propiedad intelectual, 1933), en varios aspectos:

- agrega a las obras científicas, literarias y artísticas comprenden los escritos de toda naturaleza y extensión mencionadas en el artículo 1°, los "programas de computación fuente y objeto; las compilaciones de datos o de otros materiales"
- incluye los contratos de licencia en la explotación de la propiedad intelectual del software (Art. 4°)
- indica que, para el software, el depósito que debe realizar el editor en el Registro Nacional de Propiedad Intelectual consistirá en elementos y documentos determinados por la reglamentación (Art. 57).

Esta ley modifica los artículos 1°, 4°, 9° y 57 e incorpora el artículo 55 bis a la Ley N° 11.723.

e. Ley N° 26.692 "Ley de Software" (2014)

La industria del software está considerada un sector estratégico para el desarrollo nacional, por lo que se promulgó esta ley para modificar la Ley N° 25922 "Ley de Promoción de la Industria del Software" (2004) y prorrogar el Régimen de Promoción de la Industria del Software mediante beneficios fiscales y flexibilidad para obtener certificaciones relativas a normas de calidad.

Para finalizar

La inclusión de este apartado en el apunte "Software" tiene su razón especialmente por las cuestiones relativas a derecho de autor y propiedad intelectual, no explicitadas para el software sino hasta 1994-1998, lo que supuso un vacío legal que habilitó prácticas hoy ilegales, como la piratería de software.

Podés leer más en:

- Informática legal, "La Propiedad Intelectual en la Era de Internet", <http://www.informaticalegal.com.ar/2012/10/23/la-propiedad-intelectual-en-la-era-de-internet>
- Informática legal, "Ley 11.723 – Régimen Legal de la Propiedad Intelectual" <http://www.informaticalegal.com.ar/1933/09/26/ley-11-723-regimen-legal-de-la-propiedad-intelectual/>
- Informática legal, "Decreto 165/94 – Protección para las obras de software y base de datos", <http://www.Informática++>, "Guía para elegir el sistema operativo de tu ordenador: ¿Windows, OS X o Linux?", <http://informatica.blogs.uoc.edu/2016/03/08/guia-para-elegir-el-sistema-operativo-de-tu-ordenador-windows-os-x-o-linux/>. [informaticalegal.com.ar/1994/02/03/decreto-16594-proteccion-para-las-obras-de-software-y-base-de-datos](http://www.informaticalegal.com.ar/1994/02/03/decreto-16594-proteccion-para-las-obras-de-software-y-base-de-datos)
- Informática legal, "Ley 25.036. Propiedad intelectual. Programas de computación fuente y objeto", <http://www.informaticalegal.com.ar/1998/10/14/ley-25-036-propiedad-intelectual-programas-de-computacion-fuente-y-objeto>
- Biblioteca electrónica de la AFIP, "Ley de promoción de la industria del software", http://biblioteca.afip.gob.ar/dcp/ley_c_025922_2004_08_18
- Cámara de la Industria Argentina del Software, "Ley de Software", <http://www.cessi.org.ar/sobre-ley-de-software-1363/index.html>

14. Vulnerabilidad del software

La palabra *software*, deriva de la *soft*, que en inglés significa, entre otras cosas, *blando*; esto se refiere a la característica dinámica y variable del software (incluida la información), en contraposición con el hardware. Un ejemplo de este dinamismo y variabilidad es la posibilidad de modificar, borrar, duplicar archivos, o la de configurar de diversas maneras el funcionamiento de un juego.



Ilustración 17:
<http://www.stockxpert.com/browse.phtml?f=view&id=220081>

Esta facilidad para ser modificado o adaptarse sin duda es una ventaja del software, y a la vez es su aspecto vulnerable ya que el software puede también ser modificado, borrado, o copiado por

virus informáticos, spyware, usuarios extraños a un sistema, etc.

Hay factor que propicia este tipo de vulnerabilidad: las actuales redes de computadoras, tanto pequeñas (las restringidas a una institución) como grandes redes (Internet). Las redes permiten compartir programas, e información y eso hace posibles la circulación de virus y otro tipo de intrusión en los sistemas.

Un segundo aspecto de la vulnerabilidad del software lo constituyen los medios de almacenamiento: por un lado su reducido tamaño hace fácil el transporte de información (en un CD o un *pendrive*, por ejemplo), por otro, los medios magnéticos, por ejemplo, pueden alterar su contenido si se exponen a fuertes campos eléctricos o magnéticos.

15. Conclusión

Para concluir este apunte, que espero te haya sido de alguna utilidad, no puede menos que decirse que el software es una construcción intelectual omnipresente en nuestra vida cotidiana, al punto que su desarrollo es de importancia estratégica en nuestra economía y cada vez más presente en nuestra educación. Sólo limitado por el ingenio, el software también presenta aspectos vulnerables y posibilita una cantidad de delitos que reclaman prevención y urgente legislación. El desafío es conocer para usar adecuadamente, aprender a crear con software para enriquecer nuestras habilidades, e incorporar prácticas éticas para que el software sea motor de mejor educación de la ciudadanía y mejor convivencia en sociedad.

16. Bibliografía

Algunas referencias bibliográficas se encuentran al final de cada apartado, mientras que otras, más generales, se han indicado a continuación. Llamará la atención el año de estas últimas, que se debe a que la primera edición del apunte data de agosto de 2006.

- Oliveira, Joana, *La tecnología que enseña a los robots a 'pensar' como humanos*, diario El País, consultado 20 de junio de 2017 (https://elpais.com/tecnologia/2017/06/13/actualidad/1497355219_887959.html)
- Gelbort, Roman, *Glosario de Términos de TIC*, consultado el 3 de agosto de 2006 (<http://www.argenclic.com.ar/curso/Glosario.htm>)
- Llanes Mazón, Alejandro (2005), *Glosario*, Grupo de Informática, Fac. de Biología, Univ. de la Habana, consultado en 3 de agosto de 2006 (<http://fbio.uh.cu/bioinfo/glosario.html>).
- Pequeño, Nelson (2005), *Computación aplicada a la Contabilidad, Administración y Economía. Módulo Software*, Fac. de Cs. Económicas y de Administración, Universidad de la República, Uruguay, consultado en 3 de agosto de 2006 (<http://www.ccee.edu.uy/ensenian/catcomp/material/SOFTWARE2005.pdf>).
- *Arquitectura de una computadora*, Universidad Tecnológica Nacional, Facultad Regional Rosario, Dto. Ingeniería Eléctrica, Fundamentos de Informática, consultado en 3 de agosto de 2006 (<http://www.electrica.fro.utn.edu.ar/finforma/apuntes/archivos/Tema1.pdf>).



Ilustración 18:
<https://pxhere.com/es/photo/1172012>

- *Fundamentación del Software Libre en las escuelas*, consultado el 3 de agosto de 2006 (<http://www.lugli.org.ar/wiki/bin/view/Gleduwiki/FundamentacionSoftLibreEnEscuelas>)
- Enciclopedia Britannica 2002, Inc.
- Enciclopedia Encarta '98
- *Wikipedia, la enciclopedia libre*, consultado en agosto de 2006 (<http://es.wikipedia.org>).

Ilustraciones

Ilustración 1: <https://pixabay.com/en/brain-circuit-intelligence-1845944/>.....1

Ilustración 2: <http://www.stockxpert.com/browse.phtml?f=view&id=94398>.....2

Ilustración 3: Representación gráfica de las principales ideas de este apunte.....3

Ilustración 4: https://upload.wikimedia.org/wikipedia/commons/d/d4/America_noviter_delineata.jpg.....5

Ilustración 5: <http://www.stockxpert.com/browse.phtml?f=view&id=271319>.....5

Ilustración 6: <http://www.stockxpert.com/browse.phtml?f=view&id=283085>.....6

Ilustración 7: <http://www.stockxpert.com/browse.phtml?f=view&id=174276>.....7

Ilustración 8: <http://www.stockxpert.com/browse.phtml?f=view&id=298311>.....7

Ilustración 9: Squeak-Etoys R.P.Salvador.....8

Ilustración 10: Scratch <https://www.flickr.com/photos/wfryer/9461183065>.....8

Ilustración 11: Python <https://commons.wikimedia.org/wiki/File:Python.png>.....8

Ilustración 12: Logo de Android https://www.shutterstock.com/image-photo/kyiv-ukraine-may-26-2015-android-283385381?irgwc=1&utm_medium=Affiliate&utm_campaign=FreeFoto&utm_source=39571.....12

Ilustración 13: Logo de Windows 10 <https://www.shutterstock.com/image-photo/kyiv-ukraine-may-17-2016-hand-578419504?src=pCSpzGukumvkS3CX77TMNQ-1-10>.....12

Ilustración 14: GNU y Tux bebés, mascotas de GNU y Linux <https://www.gnu.org/graphics/babies/BabyGnuTux-Big.png>.....13

Ilustración 15: <https://www.shutterstock.com/image-vector/apple-icon-332834984?src=IblmL5ecIP2sHy3h13ZFda-1-1>.....15

Ilustración 16: <http://www.stockxpert.com/browse.phtml?f=view&id=314444>.....18

Ilustración 17: <http://www.stockxpert.com/browse.phtml?f=view&id=220081>.....20

Ilustración 18: <https://pxhere.com/es/photo/1172012>.....21

Este documento fue editado con OpenOffice 2.0/LibreOffice 5.1

Mg. Ricardo P. Salvador
27 de septiembre de 2017



Atribución-NoComercial 4.0 Internacional (CC BY-NC 4.0)

<https://creativecommons.org/licenses/by-nc/4.0/>

