**El Software Libre**

Stevenson Jorkaed Arias Sanabria.

Stefanny Yulieth Oliveros Mostacilla.

Michelle Valentina Quintero Márquez.

Universidad de Pamplona. Facultad de Ingenierías y Arquitectura.

Ingeniería de Sistemas.

Ing. Fanny Casadiego Chiquillo.

14 de 12 del 2022.

**Tabla de contenido**

[Introducción 3](#_Toc121866998)

[Software libre 4](#_Toc121866999)

[1.1 ¿Qué es el Software libre? 4](#_Toc121867000)

[1.1.1 Definición. 4](#_Toc121867001)

[1.1.2 Origen. 4](#_Toc121867002)

[1.1.3 El Software libre y El Open Source. 4](#_Toc121867003)

[1.1.4 ¿Cuándo es libre un software? 4](#_Toc121867004)

[1.1.5 Licencias. 5](#_Toc121867005)

[1.1.6 Ventajas y Desventajas. 5](#_Toc121867006)

[1.1.7 Los 10 requisitos generales de software de código abierto. 6](#_Toc121867007)

[Discusión o conclusiones 9](#_Toc121867008)

[Referencia 10](#_Toc121867009)

# Introducción

El software libre tiene su inicio al principio de la década de los 80, aunque en los años 1958 al 1960 todo software producido por académicos e investigadores corporativos, eran distribuidos ampliamente y los usuarios podían modificarlo a su preferencia, por lo que técnicamente eran “software libre” pero aun este concepto no se definía.

A finales de los años 60 con la evolución de los sistemas operativos y compiladores de lenguajes de programación, los costos aumentan y algunas compañías como IBM comercializan el software incluyéndolo en el precio de hardware. A finales de los años 70 y principios de los 80 se empieza a comercializar directamente el software, por consiguiente, las compañías imponen restricciones legales a uso de todos sus nuevos códigos colocándoles derechos de autor y marcas registradas.

En el año 1983 Richard Stallman cree el proyecto GNU (que deriva de “GNU's Not Unix”, que significa no es UNIX), con este proyecto que fue motivado por no poder reparar una impresora porque no le revelaron el código de la misma. El proyecto GNU inicia una revolución hacia la obtención del software libre buscando lograr un sistema operativo que fuese distribuido entre los usuarios sin restricciones.

Para la segunda mitad de 1990 las compañías basadas en la web comienzan a operara con software. En 1997 Eric Raymond publica la catedral y bazar donde analizan la cultura hacker, esto impulsa a “Netscape Communications Corporation” a liberar en internet su suite, Netscape Communicator y ello lleva a otras compañías a potencializar el software libre.

# Software libre

## 1.1 ¿Qué es el Software libre?

### 1.1.1 Definición.

Podemos considerar el software libre como la utilización de códigos de programación que pueden ser estudiados y modificados para utilizarlos y redistribuirlos con cualquier cambio sin restricción alguna.

### 1.1.2 Origen.

La concepción del software libre tiene un largo proceso iniciado por Richard Stallman y su organización “[Free Software Foundation](https://es.wikipedia.org/wiki/Free_Software_Foundation)” en la década los 80, que proponía la libertad del usuario para modificar o adaptar el software. Siendo esto más una idea filosófica de su creador ceñida a sus ideales sobre la libertad del conocimiento el cual debe pertenecer a la humanidad sin costo alguno, es decir, él ve software como algo que pertenece a las personas y no debe ser tratado como una mercancía.

### 1.1.3 El Software libre y El Open Source.

Entre el software libre y open source (código abierto) existe una gran diferencia, aunque parezcan semejantes, el software se base en lo ético y moral, mientras que el código abierto se basa en los aspectos prácticos y técnicos, Además, no todo software de Open Source tiene que ser libre, sin embargo, todos los [productos](https://rockcontent.com/es/blog/tipos-de-productos/) y sus derivados desarrollados en software libre deben ser libres, siguiendo la definición de libre brindada por Richard Stallman y la FSF.

### 1.1.4 ¿Cuándo es libre un software?

Richard Stallman establece que se deben cumplir 4 libertades para considerar un software como libre, empezando desde el 0:

1. la libertad de usar el programa, con cualquier propósito (uso).
2. la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a las propias necesidades (estudio).
3. la libertad de distribuir copias del programa, con lo cual se puede ayudar a otros usuarios (distribución).
4. la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie (mejora).

### 1.1.5 Licencias.

En la creación de software se generan unas licencias que permiten al creador del programa liberarlo de cualquier restricción, dándole al receptor o usuario el derecho de modificar y hacer uso del programa.

Las licencias más conocidas son:

1. GLP (Licencia Publica General).
2. BSD (Total Libertad de Software).
3. MIT (Instituto de Tecnología de Massachusetts).
4. CDDL (Licencia Común de Desarrollo y Distribución).

### 1.1.6 Ventajas y Desventajas.

Al investigar a fondo se encuentra que como la mayoría de las cosas el software libre tiene sus ventajas y desventajas las cuales de una o otra forma influyen en él.

Ventajas:

1. Permite ser copiado.
2. Tiene independencia tecnológica.
3. Ahorra el mantenimiento y renovación de la tecnología.
4. Desarrollo y mejoramiento continuo.

Desventajas:

1. Carece de garantías.
2. Exige esfuerzo individual.
3. Se requiere mayor conocimiento.

### 1.1.7 Los 10 requisitos generales de software de código abierto.

1. Redistribución gratuita: No se restringe a ninguna de las partes a la venta o redistribución del software, además no se requiere ninguna regalía o tarifa por la venta del mismo.
2. Código fuente: El programa debe incluir código fuente, y debe permitir la distribución en código fuente, así como en forma compilada. Cuando alguna forma de producto no se distribuya con código fuente, debe haber un medio bien publicitado para obtener el código fuente por no más de un costo de reproducción razonable, preferiblemente descargando a través de Internet sin cargo. El código fuente debe ser la forma preferida en la que un programador modificaría el programa. No se permite el código fuente ofuscado deliberadamente. No se permiten formas intermedias como la salida de un preprocesador o traductor.
3. Obras derivadas: La licencia debe permitir modificaciones y trabajos derivados, y debe permitir que se distribuyan bajo los mismos términos que la licencia del software original.
4. Integridad del código fuente del autor: La licencia puede restringir la distribución del código fuente en forma modificada *solo* si la licencia permite la distribución de "archivos de parches" con el código fuente con el fin de modificar el programa en el momento de la compilación. La licencia debe permitir explícitamente la distribución de software construido a partir de código fuente modificado. La licencia puede requerir que las obras derivadas lleven un nombre o número de versión diferente del software original.
5. No discriminación contra personas o grupos: La licencia no debe discriminar a ninguna persona o grupo de personas.
6. No discriminación contra los campos de actividad: La licencia no debe restringir a nadie de hacer uso del programa en un campo específico de esfuerzo. Por ejemplo, no puede restringir el uso del programa en un negocio o de ser utilizado para la investigación genética.
7. Distribución de la licencia: Los derechos vinculados al programa deben aplicarse a todos los que se redistribuya el programa sin la necesidad de la ejecución de una licencia adicional por parte de esas partes.
8. La licencia no debe ser específica de un producto: Los derechos vinculados al programa no deben depender de que el programa forme parte de una distribución de software en particular. Si el programa se extrae de esa distribución y se utiliza o distribuye dentro de los términos de la licencia del programa, todas las partes a las que se redistribuye el programa deben tener los mismos derechos que los que se otorgan junto con la distribución original del software.
9. La licencia no debe restringir otro software: La licencia no debe poner restricciones a otro software que se distribuya junto con el software con licencia. Por ejemplo, la licencia no debe insistir en que todos los demás programas distribuidos en el mismo medio deben ser software de código abierto.
10. La licencia debe ser tecnológicamente neutra: Ninguna provisión de la licencia puede basarse en ninguna tecnología individual o estilo de interfaz.

# Discusión o conclusiones

La creación del software libre permite al usuario de la tecnología una mayor interacción con los diferentes códigos existentes dándole la capacidad y libertad de usarlos, modificarlos y adaptándolos para su uso personal, a la vez que impulsa a prepararse más para alcanzar este propósito.

Se puede considerar que el software libre es en la actualidad una herramienta tecnológica de uso común entre todos aquellos que procesan la misma filosofía que Richard Stallman y hacen uso de la informática ya sea como medio de trabajo o entretenimiento.

# Referencia

de Souza. (2019, 23 noviembre). Conoce qué es un software libre y sus características elementales. *rockcontent blog*. Recuperado 14 de diciembre de 2022, de <https://rockcontent.com/es/blog/software-libre/>

GioCode. (2019b, octubre 20). *Software libre vs Código abierto | GioCode* [Vídeo]. YouTube. Recuperado 14 de diciembre de 2022, de <https://www.youtube.com/watch?v=pRbs3xTltqA>

GNU. (2022, 30 de octubre). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 02:04, diciembre 15, 2022 desde <https://es.wikipedia.org/w/index.php?title=GNU&oldid=146990490>.

Historia del software libre y de código abierto. (2022, 9 de diciembre). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 02:02, diciembre 15, 2022 desde <https://es.wikipedia.org/w/index.php?title=Historia_del_software_libre_y_de_c%C3%B3digo_abierto&oldid=147812330>.

*La importancia de la filosofía del Software Libre*. (2012, 31 julio). cnti. Recuperado 14 de diciembre de 2022, de <https://www.cnti.gob.ve/noticias/actualidad/nacionales/156-softwarelibre/993-la-importancia-de-la-filosofia-del-software-libre.html>

Llamas, J. (2020, 9 diciembre). *Historia del software*. Economipedia. Recuperado 14 de diciembre de 2022, de <https://economipedia.com/definiciones/historia-del-software.html>

*Richard Stallman: “El software libre es un asunto ético del uso de la tecnología” | Sitio Web DCC*. (s. f.). <https://www.dcc.uchile.cl/richard_stallman>

Software libre. (2022, 29 de noviembre). *Wikipedia, La enciclopedia libre*. Fecha de consulta: 01:48, diciembre 15, 2022 desde <https://es.wikipedia.org/w/index.php?title=Software_libre&oldid=147623330>.

Uriarte. (2019, 18 octubre). *Software Libre*. Enciclopedia Humanidades. Recuperado 14 de diciembre de 2022, de <https://humanidades.com/software-libre/>