Tema 8 – Ley de Propiedad Intelectual

Grupo Cerberus

Antonio Martínez

Nahiara Latorre

Manuel Romero

Ricardo Espí

Universidad de Alicante – Escuela Politécnica Superior

Contenido

[1. Registro de programas 2](#_Toc470811008)

[2. Uso de recursos(imágenes, música,…) 4](#_Toc470811009)

[3. Software libre(freeware, shareware, copyleft,…) 5](#_Toc470811010)

[4. Software privativo 7](#_Toc470811011)

[5. Nuevas formas de uso(licencias del software) 7](#_Toc470811012)

# **Registro de programas**

En nuestro ordenamiento jurídico el **software**puede ser protegido mediante la Ley de Propiedad intelectual (derechos de autor) o mediante lo establecido por la Ley de Patentes.

* 1. **Definición**

La **propiedad intelectual** de una obra literaria, artística o científica corresponde al autor por el solo hecho de crearla. Se considera autor a la persona natural que crea alguna obra artística, científica o literaria. Se presumirá autor, salvo prueba de lo contrario, a quien aparezca como tal en la obra, mediante su nombre, firma o signo que lo identifique. Cuando la obra se divulgue en forma anónima o bajo seudónimo o signo, el ejercicio de los derechos de propiedad intelectual corresponderá a la persona natural o jurídica que la saque a la luz con el consentimiento del autor, mientras este no revele su identidad. La divulgación de una obra es toda expresión de la misma que, con el consentimiento del autor, la haga accesible por primera vez al público en cualquier forma.

* 1. **Registro de la propiedad intelectual**

El **Registro de la Propiedad Intelectual** tiene por objeto la inscripción de los derechos relativos a las obras, actuaciones o producciones originales, literarias, artísticas o científicas expresadas por cualquier medio o soporte, tangible o intangible, actualmente conocido o que se invente en el futuro. El ejercicio de nuestros derechos de Propiedad Intelectual sobre el**software** supone tener en cuenta algunos elementos específicos que surgen del desarrollo y creación de un programa informático:

* Documentación de análisis.
* Programas en lenguaje fuente.
* Programas compilados.
* Documentación técnica de los programas.
* Documentación de uso (manuales de usuario, explotación, etcétera).

El **Registro General de la Propiedad Intelectual** es único en todo el territorio nacional. Está integrado por los *registros territoriales* (establecidos y gestionados por las *Comunidades Autónomas*), el *Registro Central* (dependiente del *Ministerio de Educación y Cultura*) y la *Comisión de Coordinación* (integrada por un representante de cada *Comunidad Autónoma*, un representante del Ministerio de Educación y Cultura y el titular del Ministerio Central).

* 1. **La protección mediante Propiedad Intelectual: derechos de autor**

La **protección** que se aplica con carácter general a los **programas de ordenador y bases de datos** es la contenida **en la legislación sobre Propiedad Intelectual**.

* + Ley 22/87 de 11 de noviembre de Propiedad Intelectual
  + Directiva 2009/24/CE de 23 de abril de 2009 y otras.

El **derecho de propiedad intelectual o derecho de autor** se adquiere desde el mismo momento de creación de la obra, sin que sea necesario para su obtención ningún tipo de inscripción o registro.

Para adquirir el derecho de autor, la obra debe de cumplir dos **condiciones**:

* + **Expresión en una creación concreta:** el derecho de autor NO protege ideas, ni la realización técnica, sólo su forma concreta de expresión en un medio o soporte.
  + **Originalidad**: en el sentido de ser una creación intelectual propia de su autor.
    1. **¿A quién corresponden los derechos de autor cuando la creación del software se encuentra en el ámbito de una relación laboral?**

Un software será propiedad intelectual de la [empresa](http://portaley.com/2015/04/creacion-de-empresa-definir-la-forma-juridica-de-nuestro-nuevo-negocio-2a-parte/) cuando exista una relación laboral entre la misma y el empleado, **salvo que** exista constancia en las cláusulas del contrato de la delimitación de los derechos de los empleados en el marco de una relación laboral y aquéllos otros que puedan originarse de creación es fuera de ella.

* 1. **La protección mediante Propiedad Industrial: patentes**

Los programas de ordenador no son patentables en sí mismos. En casos excepcionales, éstos pueden patentarse bajo la denominación de **Invenciones implementadas en ordenador**, debiendo reunir algunos requisitos:

* Novedad mundial, actividad inventiva y aplicación industrial.
* La utilización, para su puesta en práctica, de un ordenador, una red informática u otro aparato programable en los que la ejecución de, al menos, un programa informático produce un efecto técnico que forma parte de la solución a un problema técnico concreto.
* Si es un software asociado a un dispositivo, el software debe de ser necesario para su funcionamiento.
  1. **Formas de autoprotección**
* Siempre tiene que incluir el símbolo del **copyright ©** en la primera pantalla del programa y en los manuales, delante del nombre de los titulares y autores, precisando el año y lugar de divulgación. De esta forma, se comunica a terceros que los derechos están reservados.
* Siempre tiene que incluir las condiciones de la licencia de uso y la aceptación expresa por parte del usuario, antes de instalar el programa.
* Es recomendable usar mecanismos lógicos, mecánicos y físicos (disco, llave, etc.), al margen del programa para poder ejecutarlo.
* También es recomendable registrarlos, ya que, desde el momento de su depósito, pueden preconstituir una prueba de creación y autoría del programa. Existen varias formas de registro:
  + Inscripción en el Registro General de la Propiedad Intelectual.
  + Registro en documento público o ante Notario.
* Opcional: Mantenerlos en secreto (secreto industrial).
  1. **Licencia de Software**

Una **licencia de software** es un contrato entre el licenciante (autor/titular de los derechos de explotación/distribuidor) y el licenciatario (usuario consumidor/usuario profesional o empresa) del programa informático, para utilizar el software cumpliendo una serie de términos y condiciones establecidas dentro de sus cláusulas, es decir, es un conjunto de permisos que un desarrollador le puede otorgar a un usuario en los que tiene la posibilidad de distribuir, usar y/o modificar el producto bajo una licencia determinada. Además, se suelen definir los plazos de duración, el territorio donde se aplica la licencia (ya que la licencia se soporta en las leyes particulares de cada país o región), entre otros.

Las **licencias de software** pueden establecer entre otras cosas: la *cesión de determinados derechos del propietario al usuario final sobre una o varias copias del programa informático*, *los límites en la responsabilidad por fallos*, *el plazo de cesión de los derechos*, *el ámbito geográfico de validez del contrato* e incluso pueden *establecer determinados compromisos del usuario final hacia el propietario*, tales como la no cesión del programa a terceros o la no reinstalación del programa en equipos distintos al que se instaló originalmente

# Uso de recursos (imágenes, música, …)

# Software libre (freeware, shareware, copyleft, …)

El término **software libre** se refiere el conjunto de software que, por elección manifiesta de su autor puede ser copiado, estudiado, modificado, utilizado libremente con cualquier fin y redistribuido con o sin cambios o mejoras.

* 1. **Las cuatro libertades**

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

* **Libertad 0:** la libertad de ejecutar el programa como se desea, con cualquier propósito.
* **Libertad 1:** la libertad de estudiar cómo funciona el programa y cambiarlo para que haga lo que el usuario quiera. El acceso al código fuente es una condición necesaria para ello.
* **Libertad 2:** la libertad de redistribuir copias para ayudar a su prójimo
* **Libertad 3:** la libertad de distribuir copias de sus versiones modificadas a terceros. Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

Un programa es software libre si los usuarios tienen todas las libertades. Así pues, los usuarios deberían tener la libertad de distribuir copias, sea con modificaciones o sin ellas, sea gratis o cobrando una cantidad por la distribución, a cualquiera y en cualquier lugar. El ser libre de hacer esto significa que no hay que pedir o pagar permisos.

*“Software libre”* no significa que *“no es comercial”.* Un programa libre debe estar disponible para el uso comercial, la programación comercial y la distribución comercial. La programación comercial de software libre ya no es inusual; el software libre comercial es muy importante. Puede haber pagado dinero para obtener copias de software libre, o puede haber obtenido copias sin costo. Pero sin tener en cuenta cómo obtuvo sus copias, siempre tiene la libertad de copiar y modificar el software, incluso de vender copias.

* 1. **Copyleft**

Ciertos tipos de reglas sobre la manera de distribuir software libre son aceptables, cuando no entran en conflicto con las libertades principales. Por ejemplo, el **copyleft**, es la regla en base a la cual, cuando se redistribuye el programa, no se puede agregar restricciones para denegar a los demás las libertades principales. Esta regla no entra en conflicto con las libertades principales, más bien las protege.

La forma más simple de hacer que un programa sea libre es ponerlo bajo **dominio público**, sin derechos de autor. Esto permite a la gente compartir el programa y sus mejoras si así lo desean. Pero también permite que gente no tan cooperativa convierta el programa en software privativo. Pueden realizarse tantos cambios como se quiera y distribuir el resultado como un producto privativo. Las personas que reciben el programa con esas modificaciones no tienen la libertad que el autor original les dio, ya que han sido eliminadas por el intermediario.

El objetivo del **Proyecto GNU** es dar a todos los usuarios la libertad de redistribuir y cambiar software GNU. Si los intermediarios pudiesen quitar la libertad, nuestro código podría tener muchos usuarios, pero no les proporcionaría libertad. Por eso, en lugar de poner el software GNU bajo dominio público, lo protegemos con “Copyleft”. Con copyleft cualquiera que redistribuya el software, con o sin cambios, deberá de otorgar al usuario la libertad de copiarlo y modificarlo, garantizando que se mantendrán estas libertades para todos los usuarios.

Para cubrir un programa con «copyleft» se debe, en primer lugar, declarar que sus derechos están reservados (tiene copyright). Después deben añadirse unos términos de distribución, los cuales son un instrumento legal que dotará a todo el mundo de los derechos de utilizar, modificar, y redistribuir el código del programa o de cualquier programa derivado del mismo, pero sólo si los términos de distribución no son alterados. Así, el código y las libertades se hacen legalmente inseparables.

Los desarrolladores de software privativo usan el copyright para eliminar la libertad de los usuarios; por ello se usa los derechos de autor para garantizar esa libertad. El copyleft es una forma de usar los derechos de autor en un programa. No implica abandonar los derechos de autor, ya que, si se abandonasen, el uso del copyleft sería imposible.

Una forma alternativa de copyleft es la **Licencia Pública General Affero de GNU(AGPL)**, disponible en formato HTML, texto y TexInfo. Esta licencia está diseñada para programas que pueden ser utilizados en servidores, y asegura que las versiones modificadas que se utilizan para implementar servicios para los usuarios se publiquen como código fuente disponible al público.

* 1. **Ejemplos**

**Ubuntu** es un sistema operativo basado en GNU/Linux y que se distribuye como **software libre**. Su nombre proviene de la ética homónima, en la que se habla de la existencia de uno mismo como cooperación de los demás. Está compuesto de múltiple software normalmente distribuido bajo una licencia libre o de código abierto. Ofrece el sistema de manera gratuita, y se financia por medio de servicios vinculados al sistema operativo y vendiendo soporte técnico. Además, al mantenerlo libre y gratuito, la empresa es capaz de aprovechar los desarrolladores de la comunidad para mejorar los componentes de su sistema operativo.

# Software privativo

Se denomina **software propietario, o privativo**, al software del cual no existe una forma libre de acceso a su código fuente, el cual solo se encuentra a disposición de su desarrollador y no se permite su libre modificación, adaptación o incluso lectura por parte de terceros. El término ha sido creado para designar al antónimo del concepto de software libre.

La persona física o jurídica (compañía, corporación, fundación, etc.), al poseer los derechos de autor sobre un software, tiene la posibilidad de controlar y restringir los derechos del usuario sobre su programa, lo que en el software no libre implica por lo general que el usuario solo tendrá derecho a ejecutar el software bajo ciertas condiciones.

**Adobe Creative** Cloud es un ejemplo de software privativo. Al adquirir la licencia, te da acceso al uso de varias herramientas de diseño gráfico y postproducción de Adobe (como *Illustrator*, *Photoshop* *o After Effects*), cuyas licencias por separado adquirirían un precio mayor. Al pagar por la licencia ganas derecho al uso del software adquirido según lo que se especifique en los términos y condiciones de uso (contrato), pero no podrá ser redistribuido ni modificado.

# Nuevas formas de uso (licencias del software)

El auge de las tiendas de aplicaciones ha cambiado todo el concepto de envío de software. Atrás quedan los días en los que las compañías de software enviaban a los usuarios un CD por correo. Muchos ordenadores hoy en día ni siquiera tienen lector de CD-ROM y los Smartphones no han experimentado nunca ningún tipo de descarga física.

Hay tres tipos de licencias que abundan y destacan en esta (relativamente) nueva tendencia de las tiendas de software digitales. Estas son:

* **Las licencias freemium (o lite)**, que te permiten usarlas gratis bloqueando una serie de características o ventajas para aquellos que paguen por la versión completa de la app o por micro pagos dentro de la aplicación.

Las aplicaciones freemium o lite, aportan una parte de sus funcionalidades gratuitas, pero cobra para poder utilizar el resto. Este tipo de licencia tiene dos tendencias.

Los pagos se realizan dentro de la aplicación. Por ejemplo, en el juego de los Simpson, para poder acelerar construcciones y desbloquear cierto contenido se necesitan rosquillas. Esas rosquillas se pueden comprar con dinero real. Es una estrategia muy utilizada.

* + Hay dos versiones: una gratis y una de pago. Por ejemplo en Spotify móvil, la versión gratuita no te permite elegir canción ni hacer listas reproducción, y entre otras desventajas tiene publicidad. La versión de pago prescinde de todas estas restricciones y retira la publicidad.
* **Las licencias gratuitas**, que no piden ningún tipo de remuneración al consumidor, a cambio de mostrarle publicidad dentro de la app. Hay algunas aplicaciones sin ánimo de lucro que no tienen publicidad.
* **Las licencias de pago,** que cobran por su descarga y/o su uso.

Las aplicaciones de pago cobran al usuario para poder ser descargadas. En muchas apps si se intenta adquirir sin pagar por medios ajenos a la tienda de aplicaciones, esta da error al ser ejecutada.

Las aplicaciones con licencia gratuita aportan beneficios a través del uso de publicidad. Esta publicidad puede ser más o menos invasiva, y se presenta de varias formas:

La app puede tener su propio sistema de publicidad, que la incluye de manera natural en el funcionamiento de la misma. Dos ejemplos son Twitter y Facebook. Twitter incluye la publicidad como twits marcados con la etiqueta de “Promocionado”, y Facebook la incluye en forma de posts con la etiqueta “Publicidad”.

La mayoría de apps gratuitas con publicidad se hacen servir de banners para introducir los anuncios dentro de la aplicación. Estos banners suelen ser de dos tipos. Unos ocupan toda la pantalla y muestran una imagen o vídeo publicitario, pero al agotarse un contador de tiempo este se puede cerrar, y solo aparece en ciertas ocasiones. Hay otro que está siempre (o casi siempre) presente, y suele ocupar la parte inferior de la pantalla en la que se van turnando varios anuncios.

Luego están las aplicaciones gratuitas de verdad, no utilizan publicidad y (en teoría) no aportan beneficio directo al desarrollador. Un ejemplo es Wallfer, una aplicación nacida para emular el Informer de la UA en Facebook.

Hace ya tiempo, las aplicaciones y juegos que podías disfrutar en el móvil venían instaladas de fábrica o había que descargarlas (nunca gratuitamente) utilizando los servicios WAP del teléfono. Con la llegada de los smartphones y de las tiendas de aplicaciones (como Google Play, App Store y Windows Store), todo eso cambia; se estandariza el sistema de descarga e instalación de aplicaciones, y se facilita en gran medida las condiciones de su adquisición.

La aparición de estas tiendas de aplicaciones llegan de la mano del asentamiento en el mercado de los sistemas operativos móviles más importantes: IOS ,Android, Windows Phone y BlackBerry.

El App Store se perfila junto al de Google Play como los dos más importantes mercados actuales de software. Prácticamente se podría decir que, si no está en el AppStore o no es accesible como una WebApp, no es un software que llegue al gran público.

La publicación en el AppStore puede ser un tanto complicada, ya que implica varios pasos por parte del desarrollador. A continuación, explicamos una perspectiva general de todo el proceso. Registrarse como Apple Developer

* El primer paso es registrarse como desarrollador de Apple desde http://developer.apple.com/devcenter/ios/index.action pudiendo enlazar nuestro Apple ID existente a una cuenta de desarrollador o crear una nueva.
* Una vez rellenados nuestros datos y verificada la cuenta, deberemos entrar dentro de ella.
* Entrando en el programa iOS Developer Program

Cuando estemos dentro de nuestra cuenta de Apple Developer, entraremos en nuestra cuenta y veremos en el panel de control la opción Join the Developer Program que nos dará acceso a programar apps para iOS (hay que prestar atención a ésto ya que cada tipo de app/aplicación Apple es diferente y no es el mismo este programa que el de aplicaciones de MacOS X)

Durante este proceso deberemos prestar atención a varios puntos:

En el resgistro, nuestro nombre, apellidos o cualquier otro dato no deberá contener tildes o si no recibiremos una llamada desde Apple diciendo que nuestra cuenta no puede ser completada hasta que los eliminemos del proceso de registro.

Deberemos pagar la cuota anual de desarrollador de Apple, existen varios planes según si por ejemplo queremos que nuestras apps se distribuyan sólo a través del Apple Store o también queremos que puedan ser instaladas directamente en el dispositivo sin pasar por él. En nuestro caso vamos a elegir la cuota básica que ronda unos 99$.

Deberemos aportar los datos acreditativos de empresa en el proceso o como individual, esto varía según el país

Tras realizar el pago, el proceso de registro quedará en espera para que Apple valide el nuevo usuario. Esta espera puede ser de unos pocos días, durante los cuales Apple puede ponerse en contacto con nosotros por teléfono para verificar o pedir aclaraciones sobre algún dato.

Instalando Xcode: entorno de desarrollo

En estos momentos de espera es un buen momento para instalar el entorno de desarrollo Xcode (tipo Eclipse, Netbeans, Visual Studio, ...) desde https://developer.apple.com/xcode/ que dependerá de las últimas versiones de MacOS X (no se había dicho antes, pero sí, para programar apps de iOS necesitamos un MacOS X).