Parte 3. Aplicaciones Libres

- Introducción
- Orígenes del Software
- Instalación de software y su código fuente
- Aplicaciones libres
- Créditos

Copyright 2026

Introducción

Dentro del ecosistema del Software en general, nos encontramos con la parte más visible del mismo: las aplicaciones. Las aplicaciones son los programas informáticos con los que interactuamos con el ordenador (o dispositivo) y son, al fin y al cabo, la esencia del uso o necesidad de las nuevas tecnologías.

ecosistema de aplicaciones
image 4.1.1 - Ecosistema de Aplicaciones

Nos van a facilitar la realización de tareas complejas. A diferencia del sistema operativo, cuyo objetivo es gobernar el hardware y poner orden en la ejecución de los diferentes programas, las aplicaciones realizan funciones específicas. Gracias a ellas, podemos disponer de un centro ofimático, visualizar y editar imágenes, navegar por internet, hacer cálculos matemáticos complejos (o sencillos...). Existen aplicaciones sencillas o auténticas herramientas/suites complejas realizadas con miles o millones de líneas de código

Las aplicaciones no dejan de ser un **programa escrito en un lenguaje de programación** que realiza unas funcionalidades deseadas, pero podremos disponer de Apliacaciones Libres o Aplicaciones No Libres, Comerciales o No comerciales, de código abierto pero no libres (con una licencia restrictiva por ejemplo), gratuitas o de pago, y además compiladas (o disponibles) para ciertos sistemas operativos (y otros no) o incluso para solo algunas arquitecturas concretas (por ejemplo 64 bits)

En nuestro caso nos interesa profundizar en las aplicaciones libres. Las **razones**, tal como hemos hablado al principio son aplastantes en contraposición a las aplicaciones comerciales, y más en el ámbito de la educación:

- El código es **auditable**, garantía de que el código hace lo que tiene que hacer y no realiza acciones en segundo plano. Podemos modificar el código. Al tener disponible el código y poder modificarlo, podemos cambiar algo que no nos guste, personalizarlo e incluso involucrar a los alumnos en dicho proceso
- Podemos distribuir el software de forma **ética** entre nuestros alumnos. El alumno puede disponer de dicho software en casa sin necesidad de "piratear" o realizar alguna inversión.
- Como norma general, el software libre está **disponible** para todas las plataformas. Por tanto, no dependemos de usar un sistema operativo concreto para poder hacer uso del mismo. Qué ocurre si el alumno tiene en casa un Linux y en el centro un Windows? O al revés? O usa MacOs?
- La excusa de que es el software que se usa **en la empresa** solo puede llegar a ser justificable en etapas de Formación Profesional (e incluso en éste caso podría llega a ser

Copyright 2026

discutible). El objetivo es que el alumno adquiera competencias tecnológicas, y las aplicaciones libres permiten dicha adquisición sin problemas

- "*Las aplicaciones propietarias funcionan mejor que las libres...*". Es completamente discutible e incluso falso en muchas ocasiones. Tendemos a confundir desconocer una apliación con que **funcione** mal. Hay ejemplos de aplicaciones libres muy consolidadas y con un funcionamiento ejemplar. Libreoffice, Gimp o Firefox son suites y aplicaciones libres Incluso la nave (Ingenuity) que llevó al Robot (Perseverance) a Marte, estaba basado en Software Libre Open Source, y el código se puede consultar y rehutilizar

Terminamos ésta primera parte con una lectura necesaria sobre porqué una institución educativa debería usar y enseñar software libre por el proyecto GNU:

1. Permite **compartir** conocimiento y herramientas
2. **Responsabilidad Social** ante el posible dominio de un desarrollador o multinacionales
3. **Independencia** del producto
4. **Aprendizaje** de la propia herramienta, la ética y la práctica profesional
5. **Ahorro de costes** que puede redundar en otras áreas con necesidad
6. **Calidad**

Veamos pues, qué aplicaciones y cómo tenerlas disponibles

Licencia de Creative Commons

Software Libre: Vitalinux por Arturo Martín Romero e Ignacio Sancho Morte bajo licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License.

Orígenes del Software

Una de las claves para el uso de las aplicaciones es la **distribución**. Recordamos que una de las libertades del software libre es que puedes redistribuir copias para ayudar a otros o incluso puedes distribuir copias modificadas. Es decir, que podemos coger cualquier programa hecho bajo la licencia de software libre, y además de ser libres para poderlo modificar y mejorar, podemos posteriormente entregarlo a quien queramos y de la forma que queramos.

Repositorios

Ahora bien...cómo lo distribuimos? Existen múltiples canales y gracias a esta libertad de distribución es posible aglutinar todo el software libre en determinadas ubicaciones de Internet y distribuirlo a quien lo desee de una manera centralizada. Estas ubicaciones se denominan en el mundo linux como **Repositorios**.

Repositorios
image 4.2.1 - Repositorios de software

Un Repositorio es una especie de almacen/tienda, donde puedes consultar las aplicaciones disponibles solicitar su descarga e instalación. Para hacernos una idea del concepto, android (que también es un Linux) tiene preconfigurados unos repositorios de Google de tal forma que cuando nosotros buscamos e instalamos una nueva aplicación a través de su "Play Store", lo estamos haciendo entre el software recolectado por dichos repositorios. En el mundo Linux además contamos con las siguientes ventajas:

1. Existen **dependencias** entre aplicaciones. Muchas veces, una aplicación necesita de una o varias aplicaciones adicionales para funcionar (por ejemplo librerías). Gracias a éste mecanismo, cuando solicitemos instalar una aplicación se buscarán e instalarán las aplicaciones necesarias. Ésto aporta la ventaja, de que si una misma librería va a ser usada por varias aplicaciones...¿para que "engordar" cada aplicación con la misma librería? Simplemente se marca la dependencia, y si no la tiene (de alguna necesidad previa, la descargará e instalará por nosotros
2. **Actualización**. Los repositorios mantendrán actualizadas las aplicaciones. Es decir, si se encuentra un error o una mejora y se debe actualizar la aplicación, se actualizará en dicho repositorio, y nuestro sistema actualizará dicha aplicación porque la encontrará en una versión superior a la que tiene instalada.

Copyright 2026

3. **Seguridad**. Los repositorios vienen con un sistema de autenticación basado en firmas, de forma que establecemos una comunicación segura y los paquetes vienen firmados. Es decir, si confiamos en dicho repositorio, se nos proporcionarán las claves para verificar que estamos instalando el software de dicho repositorio, y no de otro.

Repositorios preconfigurados

Por defecto, en Vitalinux están configurados los siguentes repositorios

- Repositorios oficiales *de *ubuntu*. Gran parte del software instalado proviene de la distribución de ubuntu base que lleva Vitalinux
  - Vitalinux 1.0 basado en Ubuntu Trusty
  - Vitalinux 2.0 basado en Ubuntu Bionic
- Respositorios de **Libreoffice**, para mantener una versión de la suite ofimática más actualizada que la que viene por defecto en Ubuntu. Así, contamos con las siguientes versiones:
  - Vitalinux 1.0 con Libreoffice 6.1
  - Vitalinux 2.0 con Libreoffice 6.4 (en breve pasaremos a la 7.0)
- Repositorios de **aplicaciones comunes**, que no vienen de forma oficial en Ubuntu: Google Chrome, openjdk
- Repositorio **propio de migasfree**. En éste tenemos disponibles muchas **aplicaciones propias del proyecto** para personalizar la distribución, así como **aplicaiones de terceros** que dejamos disponibles de forma actualizada para no tener que añadir repositorios de forma manual. Por ejemplo: obs-studio, scratch, geogebra... y muchas más que veremos en capitulos posteiores.

** - Los repositorios de Ubuntu están ("congelados") en un servidor del proyecto, para controlar así las actualizaciones periódicas y asegurar el proceso de forma controlada, limitándolo a unos periodos concretos de actualización*

Podemos ver la lista de los repositorios configurados en un vitalinux si ejectuamos **Software y actualizaciones**. Repositorios Vitalinux Podemos observar los repositorios comentados. *NO es recomendable eliminar dichos repositorios para preservar la integridad del sistema. De hecho en Vitalinux se recrean en cada actualización por seguridad*. También podemod añadir un volumen (repositorio local en base a un CD, en lugar de la nube) o un repositorio (Añadir...), pero recomendamos lo siguiente.

Como añadir nuevos respositorios

Muchas veces nos puede interesar añadir otras fuentes de software ya que en las que tenemos no está el software requerido o está en una versión que no es la que necesitamos. Es muy importante tener en cuenta que:

1. Se deben usar siempre un **repositorio de confianza**, para lo cual haremos referencia al propio desarrollador del software. El uqe el software sea libre, garantiza no tener que andar buscando software pirata que puede contener código malicioso
2. También puedes consultar información al grupo de soporte del programa vitalnux (***soportevitalinux@educa.aragon.es***) y ellos te añadirán el repositorio en tus equipos de forma automática, sin necesidad de hacerlo a mano (¡y en todos los equipos!). Para eso contamos con **migasfree**. Y si además es interesante para todos, se pondrá a disposición de toda la comunidad educativa.

Para añadirlo puedes usar el botón de añadir que viste en la imagen anterior. Se te pedirá la url donde se encuentra y listo, pero lo menjor es usar el comando **apt-add-repository** que nos va a permitir añadir repositorios, y si el mismo viene con una firma, se añadirá iguamente. Veamos un ejemplo. Queremos instalar obs-studio, pero no está en los repositorios (mentira, si que está, y no tendrías que hacer nada, pero es una prueba para verlo). Nos iríamos a la página de de obs oficial, y en su apartado de downloads para Linux, nos encontramos con unas instrucciones a ejecutar en una terminal (puedes abrir una en cualquier linux con la combinación CTRL+ALT+t)

```
sudo add-apt-repository ppa:obsproject/obs-studio
sudo apt update
sudo apt install obs-studio
```

Nos interesan las dos primeras instrucciones

1. Con la primera estamos indicando al sistema que **añada un nuevo repositorio**, correspondiente a obs-studio en ppa. PPA es un almacen que nos permite distribuir software personal de una forma sencilla. Admite integración contínua (cuando se libera nuevo código se generan nuevas versiones de los paquetes de software) y está muy extendido su uso en la comunidad de Ubuntu y Debian
2. Con la segunda le estamos diciendo al sistema que **actualice el listado de software disponible**...es decir, que vaya a todos los respositorios/almacenes y pregunte qué paquetes (software) hay disponbible, para luego poder instalarlos. Si el sistema no tiene un índice de paquetes disponibles, cuando vayamos a instalar un paquete no sabrá si existe ni donde está

La última linea procederá a realizar la instalación del software (y todos los paquetes/librerias que necesite). Pero eso lo veremos en breve.

Licencia de Creative Commons

Software Libre: Vitalinux por Arturo Martín Romero e Ignacio Sancho Morte bajo licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License.

Instalación de software y su código fuente

En éste apartado vamos a aprender a instalar software libre y hablaremos al final un poco sobre el código fuente.

Instalación de Software

Una vez que sabemos qué aplicación queremos usar/probar, vamos a instalarla. Para ello tendremos varios métodos

1. Vitalinux Play
2. Mediante migasfree
3. Synaptic
4. Manualmente mediante un paquete deb. Forma gráfica.
5. Línea de comandos. Instalación de aplicaciones de un repo o directamente descagando el deb.

En la mayoría de los casos, deberemos hacer uso de una cuenta con permisos de administración para poder instalar/desinstalar software. En el caso de Vitalinux por ejemplo, la cuenta de profesor y dga tienen por defecto permisos de administración.

1. Vitalinux Play

Con la finalidad de que los usuarios de Vitalinux (**docentes y estudiantes**) puedan localizar, proponer e instalar/desinstalar aplicaciones educativas, se ha creado la aplicación **Vitalinux Play**. Esta herramienta tiene catalogadas las aplicaciones más afamadas dentro del mundo educativo.
En el caso de no encontrar en este almacén alguna aplicación de la cual se tiene constancia de que esta disponible para Linux, puede informarse de ello a los técnicos informáticos del proyecto Vitalinux a través del email **soportevitalinux@educa.aragon.es** con la finalidad de que la incluyan.

Entre sus características cabría destacar:

Copyright 2026

- Permite al usuario instalar/desinstalar aplicaciones educativas **sin necesidad de credenciales**, con la finalidad de que sea un proceso lo más sencillo posible.
- Las aplicaciones se encuentran categorizadas por niveles educativos: ***infantil, primaria, secundaria, bachillerato, etc***. Esto permite filtrar las aplicaciones disponibles por el nivel educativo escribiéndolo en el campo de búsqueda.
- En la descripción de las aplicaciones se ha tratado de indicar la materia o área donde suelen ser utilizadas (***matemáticas, música, tecnología, etc***.), permitiendo de esta forma filtrarlas fácilmente escribiendo dicho patrón en el campo de búsqueda.

Para interactuar con **Vitalinux Play** simplemente debemos lanzarlo pulsando la combinación ***CONTROL+ESPACIO*** y escribir **Vitalinux Play.**

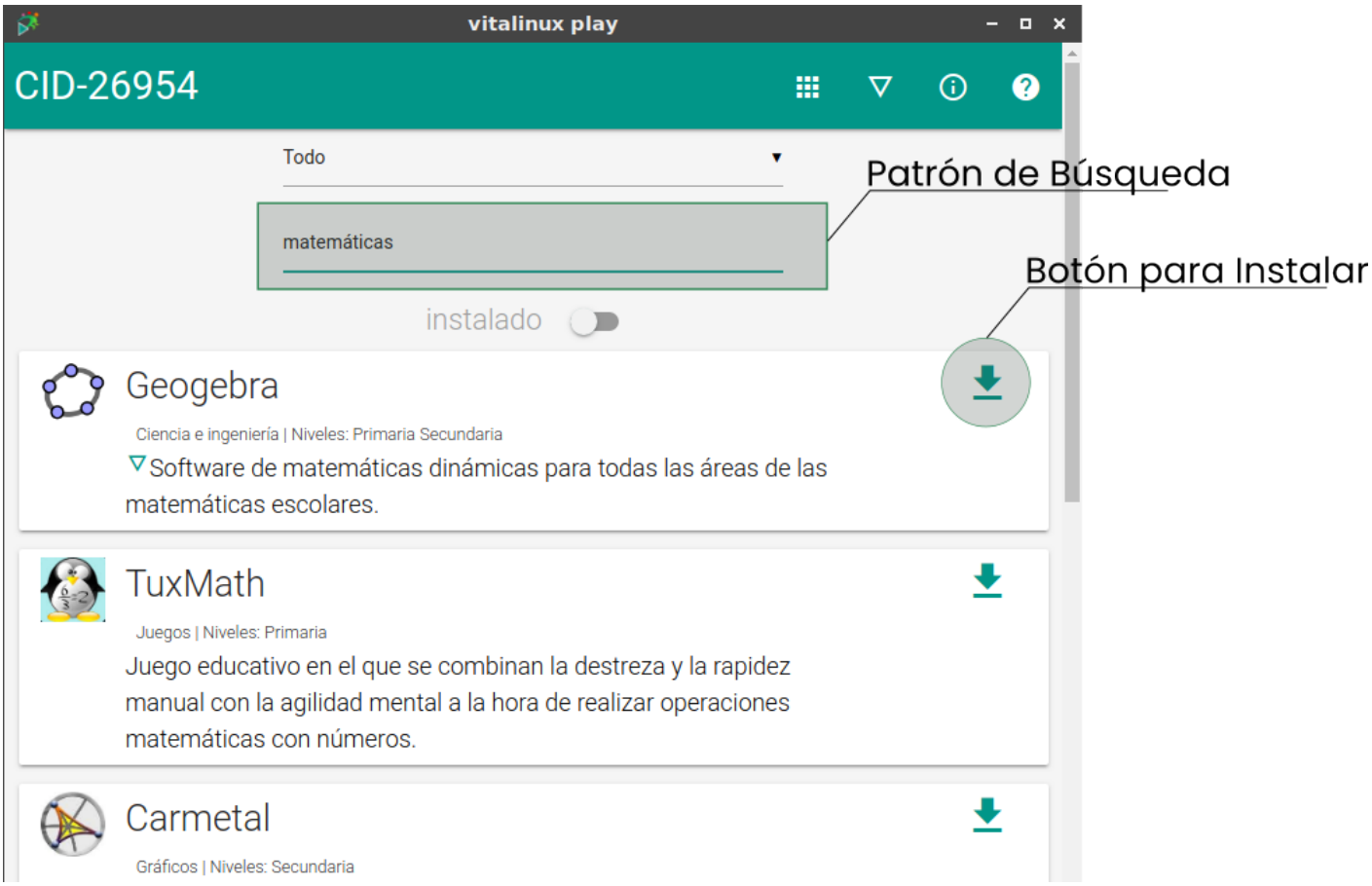


image 4.3.1., Uso de Vitalinux Play

2. Mediante migasfree

Como ya se ha dicho en varias ocasiones, una caracterísitica que vamos a encontrar en **Vitalinux** y que le diferencia del resto de distribuciones Linux actuales, es que incorpora un **cliente Migasfree** gestionado de forma centralizada y personlaziado por cada centro educativo, a demanda y de formma dinámica. En concreto, este cliente Migasfree garantiza que cada vez que

arranca Vitalinux EDU DGA y tras iniciar sesión, se establece una comunicación con el servidor Migasfree del programa de **Software Libre** de la DGA, y a través de dicha comunicación Vitalinux deja en manos de **Migasfree** la configuración del equipo en cuanto a personalización y software: Migasfree provocará en el equipo Vitalinux todo aquello que se le haya encomendado previamente por el propio centro, y en concreto:

- Qué aplicaciones deben instalarse (y si no lo están, lo hará de forma automática)
- Qué aplicaciones se tienen que desinstalar (si las encuentra instaladas).
- Actualizar las aplicaciones instaladas y del sistema
- Instalación de periféricos: Impresoras, PDIs, Webcams....
- Además realiza otras acciones como:
  - Configurar el entorno del escritorio
  - Configurar las aplicaciones: navegadores, libreoffice, arduino...
  - Limpieza del equipo
  - Gestión de usuarios

En definitiva, **cualquier aspecto que se pueda configurar vía software, se puede programar y automatizar para todos (o parte) de los eqiupos del centro**

¿Y cómo sabe qué tiene que hacer en cada equipo, si son muchos centros y cada uno usa un software diferente? A grandes rasgos, **Migasfree** tiene identificado al equipo Vitalinux en base a un identificador unívoco llamado **CID** y a un conjunto de **Etiquetas Migasfree** que se le pueden asignar. Cada centro puede personalizar así sus sistemas, y recibe la asesoría correspondiente entrando en el **programa de Asesoramiento del Proyecto Vitalinux**.

3. Synaptic

Si no encontramos la aplicación en Vitalinux Play, podemos intentar hacerlo con **Synaptic**, que buscará entre nuestros repositorios (fuentes/almacenes seguros como hemos visto antes). Aunque su aspecto visual no es tan amigable como el de otras aplicaciones equivalentes que han ido surgiendo con los años, su eficiencia y robustez han hecho que sea una aplicación muy asentada y universalmente utilizada en muchas distribuciones Linux.

Entre sus características cabría destacar:

- Permite **actualizar la lista de software disponible en los repositorios** configurados en la máquina
- Permite **buscar** rápidamente, **instalar y desinstalar** aplicaciones
- Permite solventar problemas con **paquetes rotos** (paquetes que no se terminaron de instalar correctamente)

Para interactuar con **Synaptic** simplemente debemos lanzarlo pulsando la combinación **"CONTROL + ESPACIO"**' y tecleando **"Synaptic ..."**':

Synaptic

image 4.3.2 - Gestor de Paquetes Synaptic

La pantalla de Synaptic está dividida en 4 secciones. Las dos más importantes son la lista que incluye la sección de categorías (1) en el lado izquierdo y la de los paquetes (3) en el lado derecho. Al seleccionar un paquete de la lista se mostrará una descripción del mismo (4).

Para **instalar** un paquete seleccionaremos una categoría, pulsaremos con el botón derecho del ratón sobre el paquete deseado y seleccionaremos **Marcar para instalar o bien haremos doble clic** en el nombre del paquete. Marcaremos de esta forma todos los paquetes que deseásemos instalar en el sistema y pulsaremos sobre el botón **Aplicar** para que comience su instalación. Synaptic descargará él solo los paquetes necesarios desde los repositorios

También es posible emplear el botón **Buscar** para encontrar los paquetes que deseemos instalar. Al pulsar sobre este botón podremos realizar búsquedas de programas mediante su nombre o descripción. Una vez localizado el programa que deseamos instalar hacemos doble clic sobre él para instalarlo. Si queremos eliminar un programa bastará con que pulsemos con el botón derecho encima de éste y **seleccionemos Eliminar o Eliminar por completo**.

En todos los casos, los cambios surtirán efecto una vez que hagamos clic sobre el botón de **Aplicar** .

4. Forma manual

De forma manual, dependerá de como nos faciliten el software:

- O bien es un paquete que debemos "instalar" antes de poder usar. Como lo que hemos visto hasta ahora, pero de forma manual. Serán archivos **.deb**
- Es una aplicación que tiene todo los binarios y librerías ya compilados y empaquetados, de forma que se puede lanzar directamente. En éste caso, solo debemos marcarlos para ejecutar y lanzarlo. Es lo que se conoce comúnmente como software portable. El más común actualmente es en formato **AppImage**

> “ Si optamos por la fórmula manual, debemos ser cuidadosos y saber de dónde estamos descargando el software para garantizar la seguridad e integridad de nuestro equipo y sobre todo, respetar las licencias de uso para respetar la

> legalidad del uso de la misma

Veamos un ejemplo de cada:

4.1 Paquetes deb

Descarga Software

image 4.3.3 - Descargando Software

1. Buscamos el software a instalar. Encontramos el fichero a descargar. El nombre del archivo (nootka_1.7.3-beta3_amd64.deb) nos está indicando la versión (1.7.3) y en éste caso la arquitectura (64 bits). Deberemos descargar éste si nuestra arquitectura del sistema operativo es 64 bits (lo mas común a día de hoy) o buscar el de 32 (en éste caso no hay posibilidad). También podemos encontrar el final con un all en lugar de amd64, lo que indica que sirve para cualquier arquitectura

> Observar que el software tiene versiones para múltiples arquitecturas...como debe ser :-)

Instalando Software

image 4.3.4 - Instalando Software

1. Una vez descargado, podemos instalarlo simplemente haciendo doble click sobre el paquete. Al instalarlo nos puede saltar una advertencia indicando que dicho paquete ya está en un "canal de distribución" (como es el caso), es decir, que se encuentra disponible en los repositorios...por si queremos instalarlo de ésta forma y asegurar que viene firmado y estable a la versión de distribución que tenemos (por ejemplo, las posibles librerías que necesite estarán en la versión esperada).

Es posible también encontrar url's en páginas web que directamente apuntan al paquete deb mediante un protocolo para instalar, siempre que sigamos el link desde un Linux. Lo podremos probar en la siguiente parte.

4.2 Ficheros AppImage

En el caso de ficheros AppImage, procederemos a descargarlo igualmente. En éste ejemplo, el nombre sería nootka-1.7.3-beta3-b3917-x86_64.AppImage, indicando que es para una arquitectura de 64 bits

Ésta forma de distribución, no requiere de ninguna instalación para ejecutarse, pero:

- Es muy **cómodo** ya que no requiere instalación, dependencias de librerías con versiones que no tenemos
- No requiere permisos **privilegiados** para poder usarlo (dependiendo de las acciones que quiera realizar). Los paquetes deb requieren permisos de superusuario para ser instalados.
- **No** se mantendrá **actualizado**
- Es posible que le cueste arrancar algo más, ya que contiene todas las librerías y programas adicionales que necesite
- Deberemos añadir el **lanzador** que queramos para poder tenerlo accesible de forma cómoda (menú, escritorio, lanzador de aplicaciones...)

Permisos de ejecución

image 4.3.5 - Permisos de ejecución sobre appimage

Es importante que demos **permisos de ejecución** al fichero descargado para poder ejecutarlo. Una vez hecho ésto, podremos lanzarlo con doble-click sobre el archivo descargado

AppImage en vx
image 4.3.6 - Lanzando AppImage en Vitalinux

> En vitalinux se ha facilitado su ejecución, de forma que no es necesario dar permisos de ejecución

Hay otras tecnologías de distribución de software como pueden ser Snap y Flatpak, que se están abriendo camino.

5. Línea de comandos

Éste contenido excede a los contenidos del curso, pero se muestra por si se quiere profundizar.

Mediante la línea de comandos, podemos instalar software, desinstalar, comprobar, buscar...usando los repositorios que tenemos configurados. Antes ya hemos visto algo con obs-studio. Veamos un ejemplo mas completo de acciones que puedes hacer

Actualizar la lista de paquetes disponibles en el repositorio

```
sudo apt-get update
```

Actualizar todos los paquetes instalados en el equipo

```
sudo apt-get upgrade
```

Instalar un paquetes:

```
sudo apt-get install nombre_del_paquete
```

Desinstalar paquetes:

```
sudo apt-get remove nombre-del-paquete
```

Desinstalar un paquete y sus archivos de configuración asociados:

```
sudo apt-get remove --purge nombre-del-paquete
```

Código fuente

Éste contenido excede a los contenidos del curso, pero se muestra por si se quiere profundizar.

Para finalizar éste apartado, debemos destacar un aspecto importante del software libre: es **código abierto**, y por tanto podemos consultar su código fuente, modificarlo...

> Recordar que OpenSource no siempre es software libre. Podemos tener programas de código abierto (opensource) con licencias restrictivas de uso.

Podemos descargar el código fuente usando el siguiente comando, siempre que **tengamos en los repos** marcado que queremos tener disponible además del software (deb) el código fuente (**deb-src**)

```
apt source nombre_paquete
```

Pero una forma más interesante para poder consultar el código es ir a las plataformas que se usan para la gestión de proyectos de software. La mayoría usa un sistema de control de versiones llamado git. Como servidores, hay muchos, pero los más usadas son:

- Launchpad. Por ejemplo del paquete Cheese
- Github. Aquí encontramos por ejemplo el código del software comentado antes obs-studio
- Gitlab. Por ejemplo el código de Inkspcape

Está fuera del alcance de éste curso manejar éstas herramientas de control de código. Conocer de algunos proyectos/aplicaciones el código fuente, o por lo menos seguir su desarrollo nos va a permitir involucrarnos mas en dicho programa, estar atento a las novedades, poder reportar incidencias o incluso colaborar en su mejora. Dependiendo de la edad de los alumnos y su compromiso puede ser interesante incluso hacerles participes a los mismos. Por ejemplo...podríamos ayudar a traducir un software al castellano como han hecho algunos COFO's en sus centros...estaríamos fomentando las habilidades lingüisticas y tecnológicas.

El **código fuente de los programas que se escriben para Vitalinux**, lo podemos encontrar en su propio servidor de gitlab

Para terminar, puedes visualizar un vídeo sobre la gestión del Software en Vitalinux, donde se habla del uso de las diferentes herramientas descritas

https://www.youtube.com/embed/8tBh8yz1FHY

Fuentes:

- https://itsfoss.com/remove-install-software-ubuntu/
- https://ubunlog.com/

Licencia de Creative Commons
Software Libre: Vitalinux por Arturo Martín Romero e Ignacio Sancho Morte bajo licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License .

Aplicaciones libres

En ésta última parte vamos a descubrir algunas aplicaciones libres. En Vitalinux podemos disponer de todas las aplicaciones y utilidades que encontraríamos en cualquier otro Ubuntu, y muchas otras aplicaciones que no están disponibles por defecto.

El ecosistema de aplicaciones disponibles puede llegar a ser abrumador. Descubre en los siguiente enlaces algunos ejemplos:

- Linux Apps específicas para educación ordenadas por fecha de publicación
- Blog desde Linux
- Alternativeto

Recuerda que puedes encontrar estas aplicaciones y otras muchas a través de la herramienta de gestión de paquetes Synaptic.

Veamos algunos ejemplos de aplicaciones **disponibles**, por **categorías** y para finalizar veremos **ejemplos de aplicaciones** que se usan en Aragón con la experiencia de los propios usuarios, ya que nos puede dar una aproximación mejor de dichas aplicaciones, o indicaciones de las que usan en otras comunidades autónomas.

100Free
image 4.4.1 - 100Free

Alpha Stock Images - http://alphastockimages.com/. Creative Commons 3 - CC BY-SA 3.0

1. Aplicaciones disponibles

Como Vitalinux se basa en Ubuntu, las **aplicaciones disponibles** para el sistema serán las que aparezcan en dicha rama. Pero desde el equipo de Vitalinux se ha configurado el sistema para que se **actualicen** algunas de éstas con versiones más modernas y tener disponibles **otras aplicaciones** que no están disponibles por defecto en Ubuntu:

- **Libreoffice**: Suite ofimática para procesamiento de textos, hojas de cálculo, presentaciones o edición de documentos por imágenes como publisher. Se puede poner la visualización tipo ribbon (por pestañas).

- **Navegadores** (Firefox, Chromium, Chrome, Brave, Midori...) y complementos Java actualizados.
  - Flash ha dejado de dar soporte, pero en Vitalinux se ha creado una herramienta para **lanzar recursos flash** mientras se sigan necesitando
- Aplicaciones de almacenamiento en la nube
- **Drivers para impresoras** en red y local de casi todas las marcas y modelos: HP, Epson, Canon... También con códigos
- Drivers y aplicaciones de las **PDI's antiguas** de forma cómoda para el centro. Las **MI's** ya no requieren de software adicional.
- **Aplicaciones educativas** o de terceros que no está paquetizadas por defecto para Ubuntu, pero que ahora lo están de forma sencilla...

Importante: En **Vitalinux** hay un conjunto de aplicaciones ya categorizadas (por tema y por nivel) que se han extraído del infinito conjunto de aplicaciones que hay, y que son demandadas por los centros, de forma que puede servirnos como primer filtrado. Estas aplicaciones se pueden instalar como hemos visto de forma sencilla a través de **Vitalinux Play**.

2. Aplicaciones por categorías

Veamos algunos ejemplos por **categorías**. Luego veremos ejemplos de aplicaciones que se usan en Aragón con la experiencia de los propios usuarios, ya que nos puede dar una aproximación mejor de dichas aplicaciones, o indicaciones de las que usan en otras comunidades autónomas.

Puedes clickar en el vínculo de **instalar** y te instalará la aplicación, si tienes la suerte de estar en un Linux/Vitalinux :-), pero siempre se recomienda hacerlo a través de **Vitalinux Play.**

Imagen

- GIMP. El programa de procesado de imágenes libre más usado. Instalar
- Darktable. Laboratorio fotográfico digital. Instalar
- Krita. Parecido a GIMP pero más orientado al dibujo. Instalar
- Inkscape. Dibujo vectorial libre. Sustituto de Illustrator, FreeHand, etc. Instalar
- Sweet Home-3D. Diseño de interiores 2D y 3D. Instalar
- SIR (Simple Image Resizer). Para cambiar el tamaño y otros parámetros de una imagen; admite procesado por lotes. Instalar

Copyright 2026

- Entangle. Para hacer fotografías con una cámara controlada desde el ordenador por USB, y previendo la imagen en la pantalla del ordenador. Instalar
- RawTherape RawTherape es un sistema de revelado digital para procesar imágenes RAW.

Vídeo y multimedia

- Openshot. Editor de videos en su versión más reciente. Multiplataforma (Linux, Windows y Macintosh). Instalar
- VLC. Reproductor de videos. Instalar
- Shotcut. Editor de video con algo más de potencia y posibilidades que Openshot. Multiplataforma. Instalar
- Open Broadcaster Sotware. Creación de vídeos y streaming. Instalar
- LightWorks. Estación de trabajo multimedia. Completa y compleja herramienta de produción multimedia. Nada que envidia a otras grandes. No es software libre, pero tiene una versión libre...nada que envidiar a las grandes Instalar
- Handbrake. Para ripear DVDs y convertir entre formatos de vídeo. Instalar
- Sozi. Presentaciones tipo Prezi, ver instrucciones. Instalar
- Selene. Conversor de audio y video. Instalar
- Kodi. Centro multimedia. Multiplataforma. Instalar

Sonido

- Musescore. Editor de partituras. Instalar
- Audacity. Editor de sonido más usado en educación. Instalar
- LMMS. Let's may music, vamos a hacer música. Para componer música mezclando MIDI y audio. Multiplataforma. Instalar
- Ardour. Una estación de trabajo para trabajos de Audio. Grabación, Edición y Mezcla. Muy potente. Instalar
- Mixxx. Mezclador de música para DJs, para amenizar las fiestas del cole/insti. Es multiplataforma. Instalar

Copyright 2026

- Lilypond. Lilypond es un programa de edición de partituras que sigue la tradición del grabado clásico de música Instalar
- TuxGuitar TuxGuitar es un editor y reproductor de tablaturas de guitarra multipista. Alternativa libre a herramientas como Guitar Pro. Instalar

Astronomía

- Stellarium: para enseñar lo básico sobre los planetas y constelaciones en cualquier materia (mitología, por ejemplo) y ver cómo estará el cielo esta noche o el día y la hora en que nacimos. Multiplataforma. Instalar
- Celestia: Para ver explorar el universo en casa. Multiplataforma. Instalar

Robótica y electrónica

La Robótica puede requerir de configuraciones adicionales. Se recomienda visitar ántes la página de la wiki.

- Mblock. Para programar el robot mbot. Instalar
- Web2board. Aplicación para poder programar con Bitbloq (placas por ejemplo BQ y otras muchas...). Instalar
  - Bitbloq ''offline''. Por defecto se usa la aplicación en modo ''online'' (a través de navegador, pero se puede usar de forma ''offline''). Instalar
- Fritzing. Programar para Arduino. Instalar
- Arduino. Instalar
- Scratch y Scratch 2 ''offline''. Presinstalado por defecto
- S4A. Una modificación de Scratch. Instalar
- Logisim, simulador de circuitos electrónicos digitales. Instalar

Programación

- Thonny. IDE de programación para Pyhton sencillo. Instalar
- Brackets. Editor de código para JavaScript, HTML, CSS. Instalar

Copyright 2026

- Geany. Editor de texto ligero para programación; usa bibliotecas GTK. Multiplataforma. Instalar
- Gdevelop. Para crear juegos sin saber programar. Instalar
- Netbeans. Programación en Java. Instalar
- Appinventor. Instalar
- Notepadqq Editor de texto avanzado y rápido. Muy productivo para programación. Instalar

Libros digitales

Los libros digitales suelen llevar licencias más cerradas de uso, pero desde Vitlainux se facilita su instalación (con permiso de la editorial) para no tener que instalar los libros o aplicaciones a mano...puedes ver mas información aquí.

Otras aplicaciones

- Arasuite y Araword. AraSuite es un conjunto de herramientas para trabajar con personas con discapacidades, especialemnte si estas tienen relación con la comunicación. Requiere el uso de sistemas de comunicación aumentativos que combinan pixtogramas y palabras como conceptos relacionados. Instalar
- gElemental. Tabla periódica de los elementos. Instalar
- GeoGebra: Imprescindible en matemáticas. Instalar
- Calibre. Biblioteca de ePubs ordenada, con conversión entre formatos. Instalar
- Cura. Impresión en 3D avanzada. Instalar
- SimpleScreenRecorder. Grabación de videotutoriales. Instalar
- TBO. Crea tus propios tebeos. Instalar
- Xmind. Para crear mapas conceptuales. Instalar
- Processing. Instalar
- Cmaptools. Instalar
- Scribus. Edición y maquetación de escritorio. Instalar

Coordinación TIC

- Remmina. Para abrir sesión de escritorio remoto en otro ordenador de la misma red. Instalar
- Anydesk. Para control remoto (soporte), pero que también se puede usar para controlar equipos del centro o incluso mostrar en pantalla dispositivos móviles. Viene preinstalado en Vitalinux
- Epoptes. Programa de supervisión y control de los ordenadores de un aula conectada en red. Servidor - Instalar. Cliente - Instalar
- Filezilla. Cliente de FTP. Instalar
- Transmission. Programa de descarga de torrents. Multiplataforma. Instalar
- Virtualbox. Programa de virtualización de sistemas operativos. Se pueden descargar sistemas preinstalados de [https://www.osboxes.org/ OSBoxes.org] para bajar y probar. Instalar para VX-2.0
- WireShark. Para analizar el tráfico de red. Instalar

3. Experiencia con aplicaciones

Experience
image 4.4.2 - Experience

Unsplash bank images

En la siguiente página de la documentación, podemos encontrar una tabla con Aplicaciones, clasificada por **Nivel Educativo** y **Posible Área de aplicación**, con una **descripción y experiencia** directa del propio profesor/coordinador. Reseñar que es información ya algo obsoleta.

Por otro lado, también podemos encontrar un listado categorizado de aplicaciones educativas recomendadas en las Comunidad de Madrid para su distribución MAX.

4. Equivalencias con aplicaciones conocidas

Hoy día gracias a buscadores podemos encontrar alternativas al software privado es tan sencillo como realizar una correcta búsqueda (''español e inglés'') y perder algo de tiempo en probar lo que vamos encontrando. No obstante, a modo de ejemplo indicamos las siguientes páginas:

- Alternativeto: buscador de alternativas al software privativo. Además podemos ver si es software libre o no, o qué licencia tiene.
- Jueduland: muy buena, por categorías.
- Linux Adictos: alternativas de uso cotidiano; con capturas de pantalla y breve explicación de las aplicaciones.
- Desde Linux. Recopilación de aplicaciones del 2018/2019 de éste blog de noticias.
    - Desde Linux Comparativa: Listado de comparativa con aplicaciones conocidas de otro sistema operativo.

Licencia de Creative Commons

Software Libre: Vitalinux por Arturo Martín Romero e Ignacio Sancho Morte bajo licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License.

Créditos

Autores del Curso

Autoría: Arturo Martín Romero e Ignacio Sancho Morte

Cualquier observación o detección de errores, puedes ponerte en contacto a través de soporte

Los contenidos se distribuye bajo licencia Creative Commons tipo BY-NC-SA.

Licencia de Creative Commons

Software Libre: Vitalinux por Arturo Martín Romero e Ignacio Sancho Morte bajo licencia Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional License.