

UNIVERSIDAD AUTÓNOMA GABRIEL RENE MORENO

Facultad de Ingeniería
Ciencias de la Computación y Telecomunicaciones



IMPLEMENTACIÓN UNA DISTRIBUCIÓN GNU/LINUX CON
ENTORNOS DE DESARROLLO DE SOFTWARE PREINSTALADO
PARA LOS ESTUDIANTES DE LOS PRIMEROS 5 SEMESTRES DE
LA FACULTAD DE INGENIERÍA EN CIENCIAS DE LA
COMPUTACIÓN Y TELECOMUNICACIONES

Autores:

Apaza Caseres Felix Fernando
Fernandez Lopez Jorge Gary
Maldonado Gutierrez Daniel
Ramírez Flores Juan Vladimir

Tutor:

Ing. Evans Balcazar Veizaga

Tabla de contenidos

1.	Antecedentes del problema	1
1.1.	Proyecto UREMIX	1
1.2.	Proyecto LULA	1
1.3.	Proyecto PLURIOS	2
2.	Planteamiento del Problema.	3
2.1.	Formulación de objetivos	4
2.1.1.	Objetivo general.....	4
2.1.2.	Objetivo específico	4
3.	Desarrollo.....	6
3.1.	Recolección de datos	6
3.2.	Alternativas de software libre	7
3.3.	Selección de una distribución base de GNU/Linux	8
3.4.	Implementación de la distribución FICCT-OS.	8
3.4.1.	Preparación de la distribución base.....	8
3.4.2.	Agregar los entornos de desarrollo de software.....	8
3.4.3.	Personalización de aplicaciones en la distribución base.....	9
3.4.4.	Crear un sistema operativo live de FICCT-OS.....	9
3.5.	Pruebas de instalación de FICCT-OS en máquinas virtuales y equipos físicos.....	10
3.6.	Pruebas manuales de las aplicaciones instaladas en FICCT-OS.....	11
3.7.	Liberar una versión estable de la distribución FICCT-OS.....	11
4.	Marco teórico	12
4.1.	Copia del sistema operativo.	12
4.2.	Archivos de configuración de usuario.....	12
4.3.	Sistema operativo live	12
4.4.	Software libre	13
4.5.	Licencias de software libre.....	14
4.6.	GNU/Linux.....	15
4.7.	Distribución.....	16
5.	Conclusiones y recomendaciones	18
6.	Bibliografía	19

Índice de tablas

Tabla 1. Lista de entornos de desarrollo de software en la FICCT.....	6
Tabla 2. Aplicaciones utilizadas en la FICCT y sus alternativas en software libre.....	7
Tabla 3. Cuadro comparativo de distribuciones GNU/Linux	8
Tabla 4. Prueba del instalador de FICCT-OS en VirtualBox	10
Tabla 5. Prueba del instalador de FICCT-OS en Qemu/KVM	10
Tabla 6. Prueba del instalador FICCT-OS en equipos intel.....	10
Tabla 7. Prueba del instalador FICCT-OS en equipos AMD	11

Índice de figuras

Imagen 1. Escritorio de FICCT-OS	11
Imagen 2. Mascota de la licencia GPL	14
Imagen 3. Mascota de GNU.....	15
Imagen 4. Tux, mascota de Linux.....	16
Imagen 5. Distribuciones GNU/Linux	17

Introducción

Los estudiantes de los primeros 5 semestres de la Facultad de Ciencias de la Computación y Telecomunicaciones presentan dificultades al momento de crear entornos de desarrollo de software, por tanto, el presente proyecto plantea solucionar este problema con la Implementación de una distribución GNU/Linux con entornos de desarrollo preinstalado y configurado.

Para llevar a cabo este proyecto se ha estructurado el documento en capítulos. En el capítulo I “Antecedentes del problema” mostramos proyectos realizados anteriormente tanto local, nacional e internacional, en el capítulo II “Planteamiento del problema” identificamos el problema principal, las causas y efectos, planteamos los objetivos claros para implementar este proyecto, en el capítulo III “Desarrollo” recolectamos los datos de software utilizado en los laboratorios de la FICCT y sus alternativas en software libre, seleccionamos una distribución base y describimos los pasos para crear una distribución GNU/Linux, en capítulo IV “Marco teórico” señalamos todos los fundamentos teóricos que se debe conocer para comprender e implementar este proyecto.

Palabras Clave

GNU/Linux: Aplicaciones bajo licencia GNU y el kernel Linux.

Distribución: Sistema operativo basado en GNU/Linux

IDE: Entorno de desarrollo integrado.

Entorno de desarrollo: Conjunto de aplicaciones para realizar una tarea.

FICCT: Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones

FICCT-OS: Distribución basada en Ubuntu con entornos de desarrollo de software preinstalado.

CAPÍTULO I

1. Antecedentes del problema

1.1. Proyecto UREMIX

(snifer14bs, 2011) Bueno que más que estar orgulloso de Uremix. Una Distro Boliviana

Uremix Linux Introducción Uremix Linux es una distribución Linux basada actualmente en Ubuntu desarrollada en la Universidad Autónoma Gabriel René Moreno (UAGRM) con el propósito de ser una distribución adaptada a la sociedad cruceña y boliviana. Inicios Uremix empieza como una inquietud de unos estudiantes universitarios, por aprender como es el proceso de creación de distribuciones, el nombre propuesto inicialmente dado al proyecto fue UAGReMix y el concepto de este era “una remasterización hecha en la UAGRM”, la mascota inicial fue llamada “tuxniversitario” el cual era un pingüino que llevaba puesto un gorro frigio, el cual es uno de los emblemas de la UAGRM.

1.2. Proyecto LULA

(Lozano & Blanco, 2010) Many universities use free software applications as tools for theoretical teaching and for resolving practical exercises. In many cases they are software packages in the development of which many users collaborate, in constant evolution and adaptable to different teaching needs. In Latin American universities there is a growing tendency to use it and there are communities highly committed to its development. As an attempt to compile those applications most used in Latin American universities, the LULA Project (Linux para Universidades LATInoamericanas) is about favouring the integration of this software and the exchange of educational material amongst universities. It is a non-profit initiative coordinated by the Telefónica Chair of the University of Extremadura, in which the integrating universities of the Virtual Latin American Campus (CAVILA) collaborate

1.3. Proyecto PLURIOS

(Lopez, 2021) Este es un proyecto de la empresa OpenIT – Software Libre como parte de su compromiso social para poder tener una distribución Linux acorde a las necesidades de nuestro país. La Misión es construir una metadistribución Linux localizada para Bolivia, que sea fácil de utilizar, estable y segura. La Visión del proyecto es consolidar a la metadistro PluriOS como un estándar de uso en Bolivia para entidades públicas/privadas, promover las lenguas originarias y la educación.

CAPITULO II

2. Planteamiento del Problema.

En el mundo entero, los estudiantes del área de ciencias de la computación de los primeros semestres, manifiestan tener cierto nivel de dificultad al momento de crear entornos de desarrollo software para implementar sus algoritmos y aplicaciones, los motivos son diversos y varían de región a región.

En Bolivia, especialmente en la región de Santa Cruz, los estudiantes tienen dificultad para crear entornos de desarrollo, uno de los principales motivos es, los inconvenientes para obtener los programas de instalación y el poco conocimiento que se tiene en el proceso de instalación de un software.

En la Universidad Autónoma Gabriel René Moreno en la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones se ha identificado que los estudiantes de los primeros 5 semestres, tienen una dificultad moderada en la creación de entornos de desarrollo de software. Es decir, la instalación de compiladores, intérpretes, IDE 's y las herramientas adicionales para el desarrollo de software.

Existen muchas causas para este problema que se ha identificado y de diferente índole, pero para este estudio se ha detectado que el desconocimiento de fuentes oficiales de los programas de instalación y no conocer los requisitos mínimos de software y hardware de una aplicación, son el principal problema. Influye el conocimiento básico sobre los tipos de licencia de software y la falta de experiencia en realizar una instalación de un software complejo.

Si esta situación continúa, los estudiantes tendrán largos tiempos de espera para tener un entorno de desarrollo software correctamente instalado, produciendo un deficiente aprovechamiento del

conocimiento adquirido, causando que un estudiante repruebe una materia. Los problemas en la instalación pueden ocasionar frustraciones y rechazos hacia una tecnología, también a buscar ayuda en fuentes poco confiables y seguras como blogs, canales de YouTube y páginas web.

Si los estudiantes tienen dificultad en crear entornos de desarrollo de software, se puede implementar una distribución GNU/Linux con entornos de desarrollo preinstalado y configurado.

2.1. Formulación de objetivos

2.1.1. Objetivo general

Implementar una distribución GNU/Linux con entornos de desarrollo de software preinstalado para los estudiantes de los primeros 5 semestres de la Facultad de Ingeniería en Ciencias de la Computación y Telecomunicaciones.

2.1.2. Objetivo específico

- Realizar un listado de lenguajes de programación, compiladores, intérpretes e IDE 's que se utilizan en los primeros 5 semestres en la FICCT.
- Buscar alternativas de software libre de las aplicaciones utilizadas en los primeros 5 semestres de la FICCT.
- Realizar una tabla comparativa de las aplicaciones utilizadas en la FICCT y sus alternativas en software libre.
- Seleccionar una distribución base GNU/Linux, para crear la distribución FICCT-OS con entornos de desarrollo de software preinstalado.
- Agregar y quitar aplicaciones de la distribución base.
- Implementar la distribución FICCT-OS.
- Instalar y configurar los entornos de desarrollo de software en FICCT-OS.
- Crear un sistema operativo live de la distribución FICCT-OS.

- Ejecutar pruebas del instalador de la distribución FICCT-OS en máquinas virtuales y equipos físicos.
- Ejecutar pruebas manuales de las aplicaciones instaladas en la distribución FICCT-OS.
- Realizar las correcciones necesarias para tener una versión estable de la nueva distribución FICCT-OS.
- Liberar la nueva distribución FICCT-OS con entornos de desarrollo de software preinstalado.

CAPITULO III**3. Desarrollo****3.1. Recolección de datos**

Basado en la oferta de materias 02-2021 publicada por la FICCT, se realiza un listado de materias y docentes de los primeros 5 semestres, detallando los entornos de desarrollo que utilizan cada docente y cotejando que el software se encuentre instalado en los laboratorios.

Tabla 1. Lista de entornos de desarrollo de software en la FICCT

ENTORNOS DE DESARROLLO
Lenguaje: Delphi Compilador: bcc32 Versión: 6.20 IDE: Embarcadero RAD Studio Versión: 10.2 Materia: Introducción a la informática y Programación I.
Lenguaje: Visual Basic IDE: Visual Studio 2019 Versión: 2019 Materia: Introducción a la informática.
Lenguaje: Pascal/Object Pascal Compilador: fpc Versión: 3.2.2 IDE: Lazarus Versión: 2.0.12 Materia: Introducción a la informática y Programación I.
Lenguaje: Java Compilador/Interprete: Java SE Versión: 11 IDE: Netbeans Versión: 12 Materia: Introducción a la informática y Estructura de datos I.
Lenguaje: Python Interprete: python3.8 Versión: 3.8 IDE: IDLE Versión: 3.8 Materia: Introducción a la informática y Estructura de datos I.
Lenguaje: c++ Compilador: cpp32 Versión: 6.20 IDE: Embarcadero RAD Studio Versión: 10.2 Materia: Programación II y Estructura de datos I.
Lenguaje: SQL SGBD: SQL Server 2017 DBMS: Microsoft SQL Server Management Studio 17 Versión: 17 Materia: Base de datos I y Base de datos II.
Lenguaje: prolog Interprete: prolog Versión: 8.4 IDE: SWI-Prolog Versión: 8.4

Materia: Programación lógica funcional.
--

Listado de entornos de desarrollo de software instalados en los laboratorios de la FICCT de los primeros 5 semestres.

Fuente: Elaboración propia.

3.2. Alternativas de software libre

En base al listado de aplicaciones utilizadas en la FICCT, se plantea reemplazar el software con licencia de pago, por una alternativa en software libre y multiplataforma, de manera que la creación de una distribución GNU/Linux FICCT-OS con entornos de desarrollo de software preinstalado no tenga restricciones de ningún tipo.

Tabla 2. Aplicaciones utilizadas en la FICCT y sus alternativas en software libre

Aplicaciones utilizadas en la FICCT	Alternativas de software libre
Lenguaje: Delphi Compilador: bcc32 IDE: Embarcadero RAD Studio	Lenguaje: Pascal/Object Pascal Compilador: fpc IDE: Lazarus
Lenguaje: Visual Basic IDE: Visual Studio 2019	Lenguaje: Visual Basic Compilador: gambas IDE: gambas3
Lenguaje: Java Compilador/Interprete: Java SE IDE: Netbeans	Lenguaje: Java Compilador/Interprete: OpenJDK IDE: Netbeans/Eclipse/IntelliJ IDEA Community
Lenguaje: c++ Compilador: cpp32 IDE: Embarcadero RAD Studio	Lenguaje: c++ Compilador: g++ IDE: Codeblocks
Lenguaje: SQL SGBD: SQL Server 2017 DBMS: Microsoft SQL Server Management Studio 17	Lenguaje: SQL SGBD: MariaDB DBMS: Mysql WorkBench
Lenguaje: prolog Interprete: prolog IDE: SWI-Prolog	
Lenguaje: Python Interprete: python3.8 IDE: IDLE	

Fuente: Elaboración propia.

3.3. Selección de una distribución base de GNU/Linux

La creación de una nueva distribución, se enfoca en un grupo de usuarios, con un determinado perfil, para FICCT-OS se realiza un análisis de las distribuciones más populares y con una gran comunidad de usuarios, se evalúan características puntuales para seleccionar la mejor opción.

Tabla 3. Cuadro comparativo de distribuciones GNU/Linux

	Ubuntu	Fedora	Debian	Arch
Actualizaciones	6 meses	6 meses	2 años	Diario
Soporte	5 años	13 meses	3 años	Diario
Compatibilidad	Alta	Moderada	Media	Moderada
Usabilidad	Alta	Media	Moderada	Baja
Mantenedor	Canonical/Comunidad	RedHat	Comunidad	Comunidad

Fuente: Paginas oficiales de las distribuciones GNU/Linux.

Realizado un análisis de todas las distribuciones y sus características se optó por utilizar como sistema base, la distribución Ubuntu 20.04.3 LTS, porque se adapta mejor a todas las funcionalidades y características que se quiere implementar en FICCT-OS.

3.4. Implementación de la distribución FICCT-OS.

3.4.1. Preparación de la distribución base.

En una instalación mínima de Ubuntu 20.4.3 LTS se elimina el paquete snapd, los reportes de errores y las aplicaciones de telemetría agregando mayor privacidad y seguridad a los usuarios de FICCT-OS.

3.4.2. Agregar los entornos de desarrollo de software.

Utilizando el listado de alternativas de software libre de la Tabla 2, se procede a instalar los entornos de desarrollo de software que residirán en FICCT-OS.

- ✓ Lazarus IDE v. 2.0.6+dfsg-3
- ✓ Pseint v.20210609
- ✓ Eclipse v 2021-09 (4.21.0)
- ✓ Netbeans v.12.6
- ✓ Gambas3 v.3.14.3

- ✓ rustdesk
- ✓ default-openjdk (v.11)
- ✓ CodeBlocks
- ✓ g++8

3.4.3. Personalización de aplicaciones en la distribución base.

Sistema operativo

- ✓ Instalación del paquete **libcanberra-gtk-module**
- ✓ Bloqueo paquete de **snapd** haciendo configuración en */etc/apt/preferences.d/nosnap.pref*.
- ✓ Personalización de booteo del sistema operativo live, utilizando la aplicación *ISOMASTER*
- ✓ Modificación de información en */os-release* para FICCT-OS

Entorno de escritorio

- ✓ Fondo de escritorio por defecto *FICCT-OS.svg* en */usr/share/background/*
- ✓ Logo ASCII e info de **host** en **neofetch** */usr/bin/neofetch*
- ✓ **Shell bash** personalizado con oh my bash, fix en configuración para oh my bash textura + configuración. *bashrc*
- ✓ Teclado ingles EEUU y español latinoamericano

Compiladores, interpretes e IDE

- ✓ Lazarus IDE: Apariencia similar a Delphi + lanzador en */usr/share/lazarus* + fix logo.
- ✓ Eclipse: En */usr/share/eclipse* + lanzador en */usr/share/applications* + java + ruta de **workspace** vacío (el usuario debe seleccionar su ruta).
- ✓ NetBeans: En la ruta */usr/share/netbeans* + configurar *openjdk* y habilitar java + lanzador en */usr/share/applications*.

3.4.4. Crear un sistema operativo live de FICCT-OS.

Instalado y configurado todo en la distribución base de FICCT-OS utilizando la

herramienta *systemback* se procede a crear el ISO del sistema operativo live, incluyendo

las aplicaciones instaladas y las configuraciones de usuario.

3.5. Pruebas de instalación de FICCT-OS en máquinas virtuales y equipos físicos.

La ejecución de pruebas del instalador de FICCT-OS se realizó en diferentes contextos y entornos de prueba controlado, detallamos las características de las pruebas en las siguientes tablas.

Tabla 4. Prueba del instalador de FICCT-OS en VirtualBox

Entorno de virtualización	
Software	VirtualBox
Versión	6.0.1
Disco duro	15 GB
Memoria RAM	4 GB
Procesador	Intel core i3
Dual boot	No
UEFI	No
Resultado	Instalación satisfactoria

Tabla 5. Prueba del instalador de FICCT-OS en Qemu/KVM

Entorno de virtualización	
Software	QEMU/KVM
Versión	6.2.0
Disco duro	15 GB
Memoria RAM	3 GB
Procesador	Intel core i3
Dual boot	No
UEFI	No
Resultado	Instalación satisfactoria

Tabla 6. Prueba del instalador FICCT-OS en equipos intel

Entorno de instalación	
Tarjeta madre	Intel 945
Disco duro	15 GB
Memoria RAM	3 GB
Procesador	Intel core i5
Dual boot	No
UEFI	Si
Resultado	Instalación satisfactoria

Tabla 7. Prueba del instalador FICCT-OS en equipos AMD

Entorno de instalación	
Tarjeta madre	Intel 945
Disco duro	15 GB
Memoria RAM	3 GB
Procesador	AMD
Dual boot	No
UEFI	Si
Resultado	Instalación satisfactoria

3.6. Pruebas manuales de las aplicaciones instaladas en FICCT-OS.

Con el sistema operativo instalado en una máquina virtual o equipo físico, se procede a ejecutar los entornos de programación y verificar los archivos de configuración estén correctamente creados.

3.7. Liberar una versión estable de la distribución FICCT-OS.

Ejecutadas todas las pruebas de rigor se procede a liberar la distribución GNU/Linux FICCT-OS para su uso en el entorno académico en la FICCT.

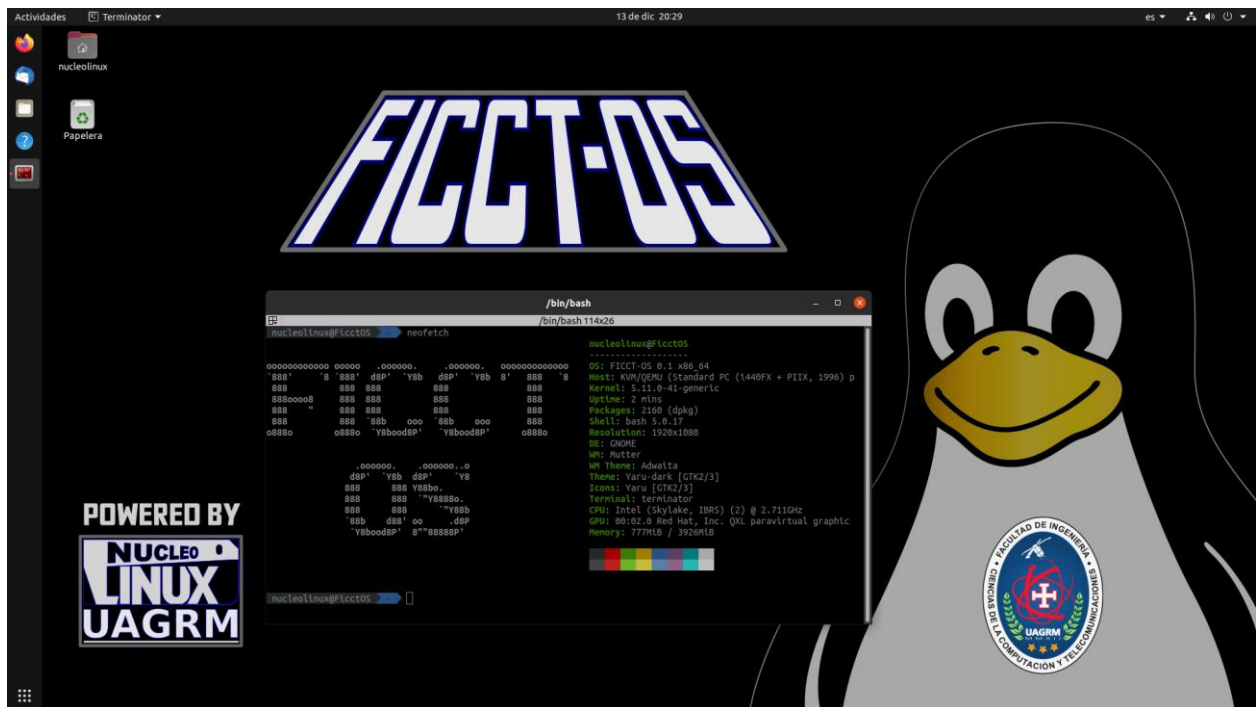


Imagen 1. Escritorio de FICCT-OS

CAPITULO IV

4. Marco teórico

4.1. Copia del sistema operativo.

(enter, 2019) Es el proceso en el que se crea una copia de los archivos importantes con el fin de poder recuperarlos en caso de una pérdida de la información. Esto es muy importante debido a que existen múltiples causas por las cuales un usuario podría experimentar este problema. Por ejemplo, la limitada vida útil de los discos duros, los robos o extravíos de los dispositivos y los ya mencionados códigos maliciosos.

4.2. Archivos de configuración de usuario

(wikipedia.org, 2021) En informática, los archivos de configuración, son archivos utilizados para configurar los argumentos(parámetros) y la configuración inicial de algunos programas informáticos. Se usan para aplicaciones de usuario, procesos de servidor y configuraciones de sistema operativo.

Algunas aplicaciones proporcionan herramientas para crear, modificar y verificar la sintaxis de sus archivos de configuración; estos a veces tienen interfaces gráficas. Para otros programas, se puede esperar que los administradores de sistemas creen y modifiquen archivos a mano usando un editor de texto; posible porque muchos son textos simples editables por humanos.

4.3. Sistema operativo live

(L., 2014) Un Sistema Live es un Sistema Operativo que corre en la memoria RAM y sin utilizar el disco duro. El hecho de no utilizar el disco tiene la ventaja de que no interfiere con el Sistema Operativo existente en el disco duro ni con los datos del mismo, De tal modo que al apagar el ordenador se evapora todo rastro de uso del mismo. Por supuesto se pierde todo aquello que no hallamos guardado debidamente, por lo que será necesario guardar

correctamente aquello que queramos, bien en un dispositivo de almacenamiento USB o bien en La Nube.

Tradicionalmente los Sistemas Live se utilizan en un CDROM o DVD, aunque los más eficientes son los instalados en PenDrive, ya que permiten mayor velocidad y proporcionan la posibilidad de guardar los datos en otra partición del dispositivo.

Para arrancar un ordenador con un Sistema Live es necesario que lo permita la BIOS. Para ello es necesario acceder al menú de arranque y escoger el dispositivo en el que tenemos el sistema. Alternativamente es preferible configurar la secuencia de arranque del BIOS para que, antes de arrancar desde el disco duro, busque un sistema en el DVD y en los dispositivos USB.

4.4. Software libre

(Free Software Foundation, 2021) «Software libre» es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el «software libre» es una cuestión de libertad, no de precio. Para entender el concepto, piense en «libre» como en «libre expresión», no como en «barra libre». En inglés, a veces en lugar de «free software» decimos «libre software», empleando ese adjetivo francés o español, derivado de «libertad», para mostrar que no queremos decir que el software es gratuito.

Con estas libertades, los usuarios (tanto individualmente como en forma colectiva) controlan el programa y lo que este hace. Cuando los usuarios no controlan el programa, decimos que dicho programa «no es libre», o que es «privativo». Un programa que no es libre controla a los usuarios, y el programador controla el programa, con lo cual el programa resulta ser un instrumento de poder injusto.

4.5. Licencias de software libre

(ecured, 2021) Se clasifica una licencia según ciertas cuestiones claves:

- ✓ Si puede ser considerada una Licencia de Software libre.
- ✓ Si es una Licencia de tipo Copyleft.
- ✓ Si es compatible con la GNU GPL (esto significa que se puede combinar un módulo que fue distribuido bajo esa licencia con otro cubierto por la GPL para hacer un programa más grande).
- ✓ Si causa cualquier otro problema particular.

Las siguientes licencias pueden considerarse de Software libre, y son compatibles con la GNU GPL:

- ✓ La Licencia Pública General de GNU, o GPL para abreviar. Es una licencia de Software Libre, y de tipo Copyleft. Es recomendada para la mayoría de los Paquetes de software.
- ✓ La Licencia Pública General Reducida de GNU, o GNU LGPL para abreviar. Es una licencia de Software libre, pero no tiene un Copyleft fuerte, porque permite que el software se enlace con módulos no libres. Sólo es recomendada para circunstancias especiales.
- ✓ Entre la versión 2 y la 2.1, la GNU LGPL cambió su nombre de "Licencia Pública General para Bibliotecas de GNU" a "Licencia Pública General Reducida de GNU", pues no es sólo para bibliotecas. Además, la GNU GPL es habitualmente más apropiada para las bibliotecas.



Imagen 2. Mascota de la licencia GPL

4.6. GNU/Linux

(Free Software Foundation, 2021) **GNU** es un sistema operativo de tipo Unix, lo cual significa que se trata de una colección de muchos programas: aplicaciones, bibliotecas, herramientas de desarrollo y hasta juegos. El desarrollo de GNU, iniciado en enero de 1984, se conoce como Proyecto GNU. Muchos de los programas de GNU se publican bajo el auspicio del Proyecto GNU y los llamamos paquetes de GNU.

El nombre «GNU» es un acrónimo recursivo de «GNU No es Unix». «GNU» se pronuncia en inglés como una sílaba sin vocal entre la g y la n. En español se puede pronunciar de la misma manera, o bien reemplazando gn por una ñ, como en ñu. También es aceptable en español la pronunciación de cada una de las letras por separado: G-N-U (ge-ene-u).



Imagen 3. Mascota de GNU

En un sistema de tipo Unix, el programa que asigna los recursos de la máquina y se comunica con el hardware se denomina «núcleo». GNU se usa generalmente con un núcleo llamado «Linux». Esta combinación es el sistema operativo GNU/Linux. Millones de personas usan GNU/Linux, aunque muchos lo llaman erróneamente «Linux».

(redhat, 2021) **Linux®** es un sistema operativo (SO) open source. En 1991, Linus Torvalds lo diseñó y creó a modo de pasatiempo. Mientras estaba en la universidad, Linus intentó crear una versión open source, alternativa y gratuita del sistema operativo MINIX, que a su vez se basaba en los principios y el diseño de Unix. Ese pasatiempo logró convertirse en el sistema operativo con la mayor base de usuarios, el más usado en los servidores de Internet

disponibles públicamente y en el único utilizado en las 500 supercomputadoras más rápidas.

Quizá lo mejor de Linux es que es open source. Linux se lanza en virtud de la Licencia de uso público GNU (GPL), lo cual significa que todos pueden ejecutar, estudiar, compartir y modificar el software. El código modificado también se puede redistribuir, e incluso vender, pero todo esto se debe hacer con la misma licencia. Esta es una de las principales diferencias con los sistemas operativos tradicionales (por ejemplo, Unix y Windows) que son propietarios, están bloqueados, se distribuyen tal como están y no se pueden modificar.



Imagen 4. Tux, mascota de Linux

Lo que se incluye y no con Linux se encuentra en constante debate. En este caso, nos referimos al kernel de Linux en conjunto con las herramientas, las aplicaciones y los servicios que lo acompañan. Todos estos elementos juntos conforman el sistema operativo funcional que la mayoría de la gente conoce como Linux. La Free Software Foundation se refiere a esta combinación como "GNU/Linux" porque algunas de estas herramientas, aplicaciones y servicios son componentes del sistema GNU. Todos estos se incorporaron al kernel de Linux, de manera tal que el sistema que conocemos es mucho más que el kernel en sí.

4.7. Distribución

(Toledo, 2020) Aunque comúnmente es utilizado como nombre para todo el sistema operativo, Linux es solo el nombre del Kernel. Es una pieza de software que maneja las interacciones entre el hardware y las aplicaciones del usuario final. Por otro lado, hablar de una Distribución de Linux, hace referencia a un sistema operativo completo, construido sobre

el núcleo de Linux. Generalmente incluye un asistente de instalación y muchas aplicaciones, preinstaladas o empaquetadas para una instalación sencilla. Muchas distribuciones de GNU/Linux se denominan solamente Linux, es importante que tengamos presente la diferencia entre estos y que el proyecto GNU es el que realmente ha aportado mucha de la estructura para el funcionamiento del mismo.



Imagen 5. Distribuciones GNU/Linux

CAPITULO V

5. Conclusiones y recomendaciones

Luego de identificar el problema y realizar la recolección de datos correspondientes, se concluye que tener un sistema operativo con entornos de desarrollo de software preinstalado, es un instrumento que facilitara al estudiante efectuar sus prácticas y aplicar el conocimiento adquirido durante el proceso de formación.

Elaborado el listado de alternativas de software libre de las aplicaciones utilizadas en los laboratorios de la FICCT y elegida la distribución Ubuntu 20.04.3 se procede a instalar y configurar los entornos de desarrollo de software, dando como resultado una nueva distribución llamada FICCT-OS.

Se ejecutan las pruebas en entornos virtuales y físicos de la imagen live de instalación de FICCT-OS de manera satisfactoria, por lo tanto, se procede a liberar la primera versión de la distribución.

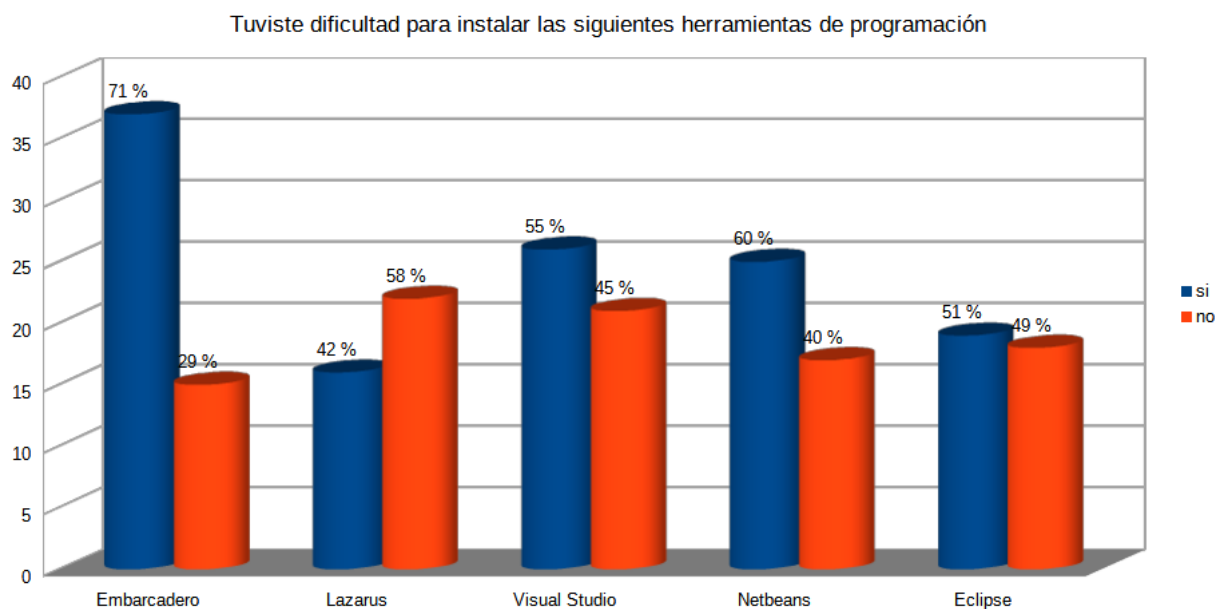
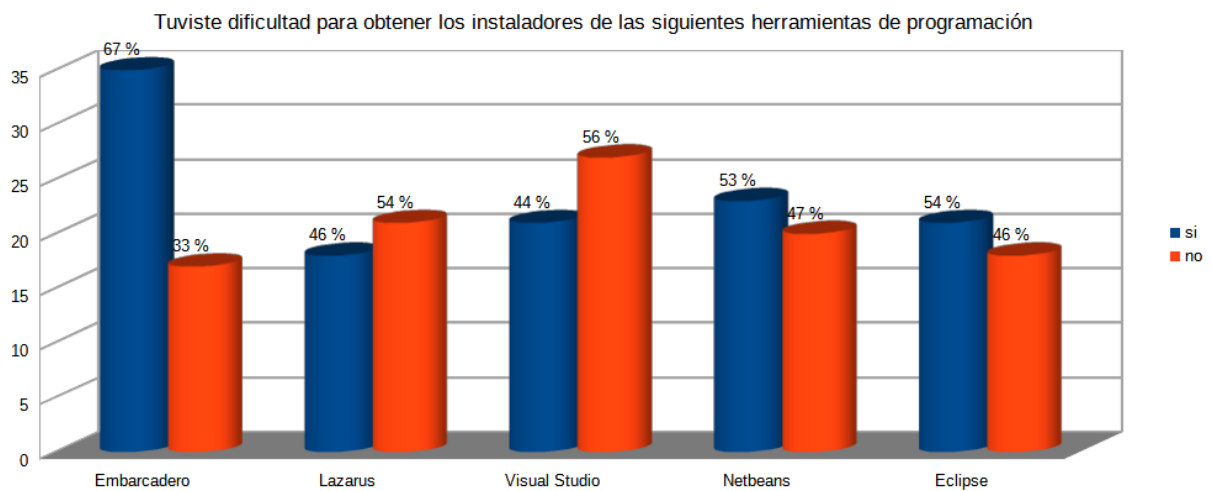
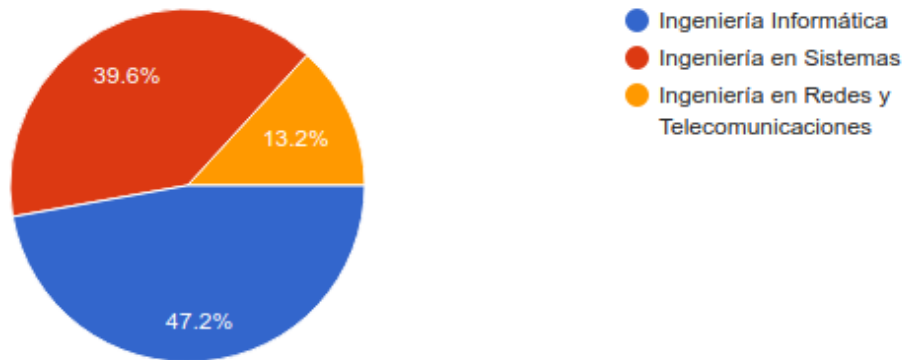
Considerando el proyecto desarrollado y en función a los datos recolectados y procesados, se formulan algunas recomendaciones a los entes educativos y administrativos.

- ✓ Promover el uso de compiladores, interpretes e IDE's multiplataforma y de licencia GPL.
- ✓ Organizar eventos para promover el uso de GNU/Linux y software libre.
- ✓ Realizar talleres para enseñar a crear entornos de desarrollo.
- ✓ Permitir a los estudiantes presentar sus proyectos utilizando tecnologías opensource y software libre.
- ✓ Utilizar a FICCT-OS como un punto de partida, para facilitar al estudiante su proceso de enseñanza/aprendizaje en la FICCT.

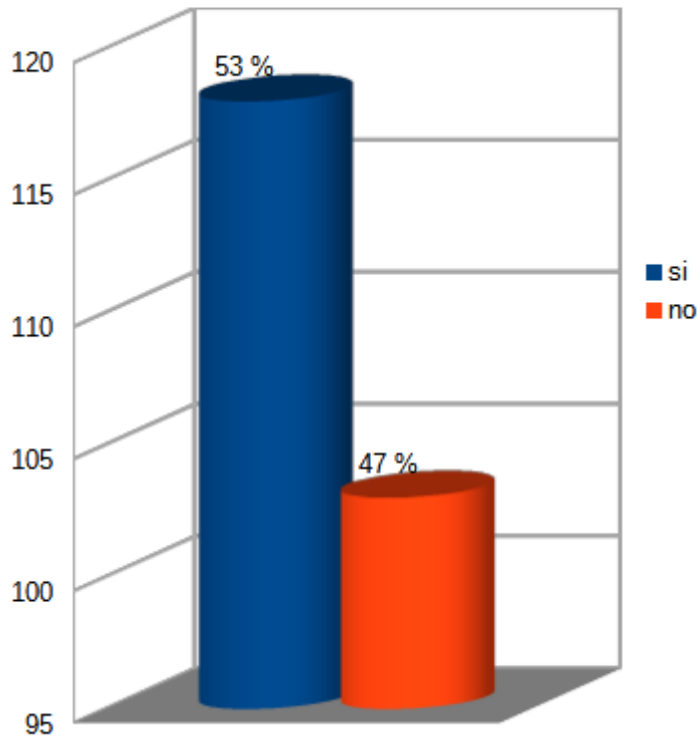
6. Bibliografía

- ecured. (2021). <https://www.ecured.cu>. Obtenido de https://www.ecured.cu/Licencia_de_software_libre
- enter. (marzo de 2019). <https://www.enter.co>. Obtenido de <https://www.enter.co/wp-content/uploads/2019/03/guia-backup.pdf>
- Free Software Foundation. (2 de Septiembre de 2021). <https://www.gnu.org>. Obtenido de <https://www.gnu.org/philosophy/free-sw.es.html>
- Free Software Foundation. (11 de diciembre de 2021). <https://www.gnu.org>. Obtenido de <https://www.gnu.org/home.es.html>
- L., A. J. (2014). <https://edu.fauno.org>. Obtenido de <https://edu.fauno.org/SistemaLive.html>
- Lopez, J. (2021). <https://plurios.openit.dev>. Obtenido de https://plurios.openit.dev/blog/?page_id=2
- Lozano, & Blanco. (14 de Abril de 2010). <http://www.ieec.uned.es>. Obtenido de <http://www.ieec.uned.es/investigacion/educon2010/searchtool/EDUCON2010/papers/2010S10D04.pdf>
- redhat. (2021). <https://www.redhat.com>. Obtenido de <https://www.redhat.com/es/topics/linux>
- sniferl4bs. (11 de Octubre de 2011). sniferl4bs.com. Obtenido de <https://sniferl4bs.com/tags/uremix/page/2/>
- Toledo, D. (septiembre de 2020). <https://www.linuxeros.org>. Obtenido de <https://www.linuxeros.org/2020/09/que-es-una-distribucion-de-linux.html>
- wikipedia.org. (2021). <https://es.wikipedia.org>. Obtenido de https://es.wikipedia.org/wiki/Archivo_de_configuraci%C3%B3n

Encuesta realizada a los estudiantes de la FICCT



Tuviste dificultad para obtener los instaladores de las siguientes herramientas de programación



Tuviste dificultad para instalar las siguientes herramientas de programación

