



UNIVERSIDAD ESTATAL DE BOLÍVAR
FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN
EMPRESARIAL E INFORMÁTICA
ESCUELA DE SISTEMAS

TRABAJO DE GRADUACIÓN, PREVIO A LA OBTENCIÓN
DEL TÍTULO EN TECNÓLOGA EN INFORMÁTICA
APLICADA.

TEMA

USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE
APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA
EN INFORMÁTICA APLICADA DE LA ESCUELA DE
SISTEMAS AÑO 2011.

AUTORA

ARÉVALO QUINABANDA FLOR BEATRIZ

DIRECTOR

ING. DARWIN PAUL CARRIO BUENAÑO

GUARANDA – ECUADOR

2012



UNIVERSIDAD ESTATAL DE BOLÍVAR
FACULTAD DE CIENCIAS ADMINISTRATIVAS, GESTIÓN
EMPRESARIAL E INFORMÁTICA
ESCUELA DE SISTEMAS

TRABAJO DE GRADUACIÓN, PREVIO A LA OBTENCIÓN
DEL TÍTULO EN TECNÓLOGA EN INFORMÁTICA
APLICADA.

TEMA

USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE
APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA
EN INFORMÁTICA APLICADA DE LA ESCUELA DE
SISTEMAS AÑO 2011.

AUTORA

ARÉVALO QUINABANDA FLOR BEATRIZ

GUARANDA – ECUADOR

2012

I. DEDICATORIA

Quiero dedicar este trabajo a Dios a mi familia, por acompañarme en cada una de las locuras que he emprendido y ser siempre mis más fervientes hinchas.

A mi “papi” Agustín por todo lo que me han dado en esta vida, especialmente por sus sabios consejos y por estar a mi lado en los momentos difíciles.

A mi “mami” Narcisa quien con su simpleza me ha ayudado a encontrar la luz cuando todo es oscuridad y por estar siempre dispuesta a ayudarme en las buenas y en las malas y siempre será mi guía en mi vida.

A mis hermanos, quienes me acompañaron en silencio con una comprensión a prueba de todo por escucharme, soportarme y convertirse en mis mejores amigos.

Aquellas personas quienes fueron mi fortaleza y apoyo en todos los momentos de mi larga trayectoria estudiantil que al final, llena de perseverancia, hoy, puedo sentirme satisfecha por el esfuerzo realizado por eso les dedico este trabajo.

FLOR

II. AGRADECIMIENTO

Me complace de sobre manera a través de este trabajo exteriorizar mi sincero agradecimiento a la Universidad Estatal de Bolívar, Facultad de Ciencias Administrativas, Gestión Empresarial e Informática, Escuela de Sistemas, y en ella a los distinguidos docentes quienes con su profesionalismo y ética puesto de manifiesto en las aulas enrumban a cada uno de los que acudimos con sus conocimientos que nos servirán para ser útiles a la sociedad.

A mi Director de Tesis Ing. Darwin Paul Carrión Buenaño quien con su experiencia como docente ha sido la guía idónea, durante el proceso que ha llevado el realizar esta tesis, me ha brindado el tiempo necesario, como la información para que este anhelo llegue a Ser felizmente culminada.

FLOR

III. CERTIFICACIÓN DEL DIRECTOR.

Ing. Darwin Paul Carrión Buenaño,
DIRECTOR DE TESIS

CERTIFICA

Que el trabajo de investigación de tesis de grado titulada “USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS AÑO 2011”, realizado por la señorita:ARÉVALO QUINABANDA FLOR BEATRIZ, ha sido debidamente revisada, por lo que autorizo su publicación para la predefensa de acuerdo al reglamento vigente.

Es todo cuanto puedo certificar en honor a la verdad, facultando a la interesada dar al presente el uso legal que estimen conveniente.

Guaranda, Julio 18 del 2012



.....
Ing. Darwin Paul Carrión Buenaño
DIRECTOR DE TESIS

IV. AUTORÍA NOTARIADA.

Las ideas, criterios y propuestas expuestos en el presente informe final para el trabajo de grado son de exclusiva responsabilidad de la autora..



ARÉVALO QUINABANDA FLOR BEATRIZ

C.I:0201966090

DR. JOSE CORDOVA NUNEZ

1 RECONOCIMIENTO DE FIRMAS Y RUBRICAS



R. DEL E.
NOTARIA
PRIMERA
San Miguel
Prov. Bolívar

2 En la Ciudad de San Miguel, Cantón del mismo nombre,
3 Provincia de Bolívar, República del Ecuador, hoy día miércoles
4 diecinueve (19) de septiembre del dos mil doce, ante mí,
5 DOCTOR JOSÉ CÓRDOVA NÚÑEZ, NOTARIO PUBLICO
6 PRIMERO DEL CANTÓN SAN MIGUEL, comparece la
7 señora: Arévalo Quinabanda Flor Beatriz, casada, con el objeto
8 de reconocer su firma y rúbrica, que obra al pie del documento
9 que antecede. Al efecto, siendo conocedor de los delitos del
10 perjurio e instruido por mí el Notario, de la obligación que
11 tiene de decir la verdad, declara y manifiesta, que la firma y
12 rúbrica, impresa en el mismo, es suya propia, la misma que la
13 utiliza en todos sus actos públicos y privados y como tal la
14 reconoce, firmando en unidad de acto, de todo lo cual Doy Fe.-

15
16
17
18
19
20



Dr. José Córdova Núñez
EL NOTARIO



0201966090

V. TABLA DE CONTENIDOS

I. DEDICATORIA	I
II. AGRADECIMIENTO	II
III. CERTIFICACIÓN DEL DIRECTOR.	III
IV. AUTORÍA NOTARIADA.....	IV
V. TABLA DE CONTENIDOS	V
VI. LISTADE CUADROS Y GRAFICOS	VII

CAPITULO I

1 MARCO REFERENCIAL.....	2
1.1 TEMA.....	2
1.2 ANTECEDENTES.....	2
1.3 FORMULACIÓN DEL PROBLEMA	4
1.4 JUSTIFICACIÓN.....	4
1.5 OBJETIVOS.....	6
1.5.1 OBJETIVO GENERAL	6
1.5.2 OBJETIVOS ESPECÍFICOS.....	6
1.6 MARCO TEÓRICO	7
1.6.1 MARCO REFERENCIAL	7
1.6.2 MARCO GEO REFERENCIAL.....	8
1.6.3 MARCO LEGAL	10
1.6.4 MARCO CONCEPTUAL.....	13
1.7 COMPROBACIÓN DE LA HIPÓTESIS	38
1.8 METODOLÓGIA	39
1.8.1 MÉTODOS	39
1.9 TIPO DE INVESTIGACIÓN.....	39
1.10 TÉCNICAS E INSTRUMENTOS PARA LA OBTENCIÓN DE DATOS	40
1.11 UNIVERSO Y MUESTRA.....	41
1.12 PROCESAMIENTO DE DATOS.....	41

CAPITULO II

2	MARCO PROPOSITIVO.....	43
2.1	USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS.....	43
2.2	PLAN OPERATIVO.....	44
2.2.1	PLAN DE UTILIZACIÓN DE SOFTWARE LIBRE PARA DESARROLLO DE APLICACIONES WEB.....	44

CAPITULO III

3	ANÁLISIS E INTERPRETACIÓN DE RESULTADOS	53
3.6	CONCLUSIONES Y RECOMENDACIONES.....	64
3.6.1	CONCLUSIONES:.....	64
3.6.2	RECOMENDACIONES:.....	65
	BIBLIOGRAFÍA	67
	WEBGRAFIA.....	67

VI. LISTA DE CUADROS Y GRAFICOS

Cuadro N° 1.- Comparaciones entre lenguajes para el desarrollo de Aplicaciones Web.	53
Gráfico N° 1 Comparaciones entre lenguajes para el desarrollo de Aplicaciones Web	54
Cuadro N° 2.- Comparaciones entre Bases de Datos para el desarrollo de Aplicaciones Web	56
Gráfico N° 2 Comparaciones entre Bases de Datos para el desarrollo de Aplicaciones Web... ..	57
Cuadro N° 3.- Comparaciones entre Editores para el desarrollo de Aplicaciones Web	58
Gráfico N° 3 Comparaciones entre Editores para el desarrollo de Aplicaciones Web.....	59
Cuadro N° 3.- Comparaciones entre Graficadores para el desarrollo de Aplicaciones Web. 60.....	60
Gráfico N° 3 Comparaciones entre Graficadores para el desarrollo de Aplicaciones Web.	61
Cuadro N° 3.- Comparaciones entre audios para el desarrollo de Aplicaciones Web.	62
Gráfico N° 3 Comparaciones entre audios para el desarrollo de Aplicaciones Web..	63

CAPITULO I

1 MARCO REFERENCIAL

1.1 TEMA.

USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS EN EL AÑO 2011.

1.2 ANTECEDENTES.

Entre los años 1960 y 1970, el software no era considerado un producto sino un añadido que los vendedores de las grandes computadoras de la época (las mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de la década de 1970, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

En 1971, cuando la informática todavía no había sufrido su gran boom, las personas que hacían uso de ella, en ámbitos universitarios y empresariales, creaban y compartían el software sin ningún tipo de restricciones.

Con la llegada de los años 1980 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar sistemas operativos

privativos, forzando a los usuarios a aceptar condiciones restrictivas que impedirían realizar modificaciones a dicho software.

En caso de que algún usuario o programador encontrase algún error en la aplicación, lo único que podía hacer era darlo a conocer a la empresa desarrolladora para que ésta lo solucionara. Aunque el programador estuviese capacitado para solucionar el problema y lo deseara hacer sin pedir nada a cambio, el contrato le impedía que modificase el software.

El mismo Richard Stallman cuenta que por aquellos años, en el laboratorio donde trabajaba, habían recibido una impresora donada por una empresa externa. El dispositivo, que era utilizado en red por todos los trabajadores, parecía no funcionar a la perfección, dado que cada cierto tiempo el papel se atascaba. Como agravante, no se generaba ningún aviso que se enviase por red e informase a los usuarios de la situación.

La pérdida de tiempo era constante, ya que en ocasiones, los trabajadores enviaban por red sus trabajos a imprimir y al ir a buscarlos se encontraban la impresora atascada y una cola enorme de trabajos pendientes. Richard Stallman decidió arreglar el problema, e implementar el envío de un aviso por red cuando la impresora se bloqueara. Para ello necesitaba tener acceso al código fuente de los controladores de la impresora. Pidió a la empresa propietaria de la impresora lo que necesitaba, comentando, sin pedir nada a cambio, qué era lo que pretendía realizar. La empresa se negó a entregarle el código fuente. En ese preciso instante, Stallman se vio en una encrucijada: debía elegir entre aceptar el nuevo software propietario firmando acuerdos de no revelación y acabar desarrollando más software propietario con licencias restrictivas, que a su vez deberían ser más adelante aceptadas por sus propios colegas.

Con este antecedente, en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo la definición de software libre y el concepto de "copyleft", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software.

En un mundo tan competitivo es necesario que el alumno conozca la mejores alternativas para favorecer su labor como profesional y el código libre presenta varias opciones que permite al estudiante desarrollar aplicaciones con facilidad ya que cuenta con varios programas además la ventaja de ser de código abierto, permitiendo desarrollar al estudiante libremente aplicaciones.

Razón por la que se realizará un estudio en la Carrera de Tecnología en Informática Aplicada, para analizar que software utilizan los alumnos y profesores en lo referente a diseño de páginas web y así establecer resultados que facilitaran la realización de mi trabajo de investigación.

1.3 FORMULACIÓN DEL PROBLEMA

¿CÓMO INFLUYE LA FALTA DE USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS AÑO 2011?

1.4 JUSTIFICACIÓN

En este trabajo se trata de una propuesta innovadora para la Universidad Estatal de Bolívar acerca del estudio del desarrollo de aplicaciones Web,

utilizando herramientas del Software Libre, se enfoca desde un punto de vista constructivista, se fomenta la reflexión sobre las relaciones entre las tecnologías de Información con carácter libre y la enseñanza, la investigación pretende complementar la formación de los alumnos de la carrera tecnología en informática aplicada, en temas relacionados a la Ingeniería Web.

Este proyecto está dirigido a realizar un análisis de la incidencia del software libre en la formación de los tecnólogos en Informática Aplicada de la Universidad Estatal de Bolívar, especialmente en el desarrollo de aplicaciones web.

Para lo cual se identifica las opiniones de los estudiantes y el docente de la carrera de Tecnología en Informática Aplicada, Esto permitirá tener una visión de cómo influye la falta del estudio de software libre en la formación académica de los estudiantes.

Se recopilará información que permitirá sustentar esta investigación, la misma que servirá para futuros estudios sobre este tema.

El presente trabajo de investigación está dirigido a mejorar la calidad de educación impartida en esta universidad, y fundamentada en el decreto Presidencial No. 1014 DECRETO DE SOFTWARE LIBRE en el cual se decreta:

Artículo 1. Establecer como política pública para las Entidades de la Administración Pública Central la utilización de Software Libre en sus sistemas y equipamientos informáticos.

Artículo 2. Se entiende por Software Libre, a los programas de computación que se pueden utilizar y distribuir sin restricción alguna, que permitan su acceso a los códigos fuentes y que sus aplicaciones puedan ser mejoradas.

Estos programas de computación tienen las siguientes libertades:

- a) Utilización del programa con cualquier propósito de uso común.
- b) Distribución de copias sin restricción alguna.
- c) Estudio y modificación del programa (Requisito: código fuente disponible)
- d) Publicación del programa mejorado (Requisitos: código fuente disponible).

1.5 OBJETIVOS

1.5.1 OBJETIVO GENERAL

- ❖ Analizar el uso de software libre para el desarrollo de aplicaciones web.

1.5.2 OBJETIVOS ESPECÍFICOS

- ❖ Investigar el uso de software libre para el desarrollo de aplicaciones web.
- ❖ Identificar los beneficios de software libre para el desarrollo de aplicaciones web.
- ❖ Motivar al desarrollo de aplicaciones web con software libre.

1.6 MARCO TEÓRICO

1.6.1 MARCO REFERENCIAL

La Universidad Estatal de Bolívar se inicia el 22 de octubre de 1977, gracias al auspicio económico del Consejo Provincial. Inicialmente funcionó como Extensión de la Universidad de Guayaquil, adscrita a la Facultad de Ciencias Administrativas, Escuela de Administración de Empresas Agroindustriales, como la primera en crearse, cumpliendo así con una de las más caras aspiraciones de la sociedad bolivarenses: contar con un centro de educación superior que atienda las demandas del desarrollo regional.

Finalmente el desarrollo académico alcanzado por la Extensión Universitaria de Guaranda fue reconocido por el CONUEP quien aprobó la creación de la Universidad Estatal de Bolívar (UEB) el 20 de junio de 1989, mediante decreto No 32 del H. Congreso Nacional. El Presidente Constitucional de la República del Ecuador en ese entonces Dr. Rodrigo Borja Cevallos, firma el ejecútese el 29 de junio del mismo año y se publica en el Registro Oficial No 225, el 4 de julio de 1989, "la creación de la Universidad Estatal de Bolívar (UEB)", actuando como primer Rector el Ing. Gabriel Galarza López.

Una vez que la Universidad Estatal de Bolívar adquirió la personería jurídica, inició la etapa de organización interna y la estructuración de propuestas para la formación de profesionales. Se crea la carrera de Enfermería en 1986, luego la de Educación Física, Tecnología Avícola y finalmente Contaduría Pública, la que más tarde cambiaría su nombre por Contabilidad y Auditoría (1990). En la UEB las actividades y funciones se normaron por sus Estatutos aprobados por el H. Consejo Universitario el 14 de julio de 1989 y por el ente regulador de las Universidades de aquel entonces CONUEP, hoy llamado

CONESUP, en donde se confiere legitimidad a la organización institucional basada en organismos, Facultades, Departamentos, Unidades Académicas y Servicios y es así que la Universidad Estatal de Bolívar actualmente funciona con cinco Facultades, siendo la última la de Jurisprudencia que fue creada el 12 de junio del 2002 . Todas ellas fueron creadas en base a los requerimientos de la sociedad y con la finalidad de buscar la calidad profesional y dar respuesta a la misión institucional y al desarrollo de la provincia.

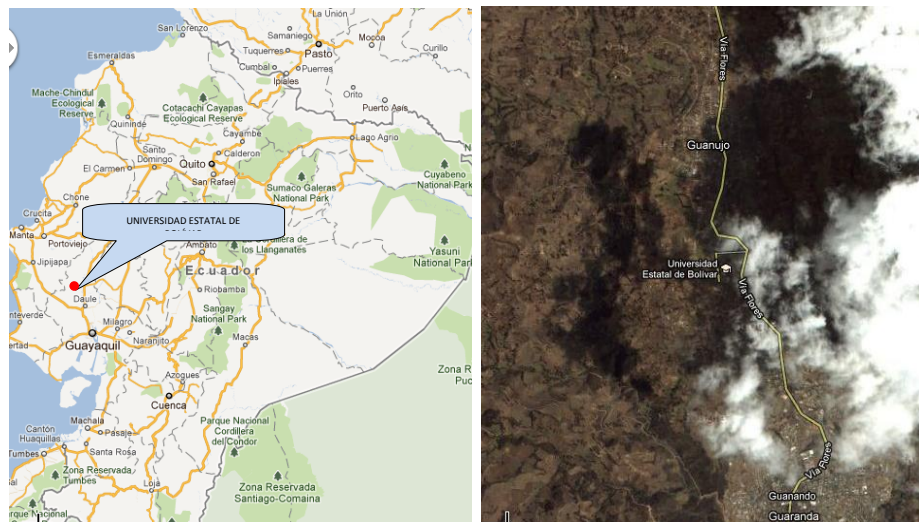
1.6.2 MARCO GEO REFERENCIAL

La carrera de Tecnología en Informática Aplicada de la escuela de sistemas se encuentra ubicada en los edificios de la Facultad de Ciencias Administrativas Gestión Empresarial e Informática de la Universidad Estatal de Bolívar localizada en el Km. 3 1/2 sector Alpachaca Cantón Guaranda, Provincia Bolívar, País Ecuador.

Longitud: 79°0'26''

Latitud: 1°34'17''

Sistema de referencia wgs84 zona 17 sur.



Fuente: <http://maps.google.com.ec/maps?hl=es&tab=wl&q=go> Fuente:

<http://maps.google.com.ec/maps?hl=es&tab=wl&q=go>



Fuente: <http://maps.google.com.ec/maps?hl=es&tab=wl&q=go>

1.6.3 MARCO LEGAL

Decreto de Software Libre No. 1014

Artículo 1. Establecer como política pública para las Entidades de la Administración Pública Central la utilización de Software Libre en sus sistemas y equipamientos informáticos.

Artículo 2. Se entiende por Software Libre, a los programas de computación que se pueden utilizar y distribuir sin restricción alguna, que permitan su acceso a los códigos fuentes y que sus aplicaciones puedan ser mejoradas.

Estos programas de computación tienen las siguientes libertades:

- a) Utilización del programa con cualquier propósito de uso común.
- b) Distribución de copias sin restricción alguna.
- c) Estudio y modificación del programa (Requisito: código fuente disponible)
- d) Publicación del programa mejorado (Requisitos: código fuente disponible)

Artículo 3. Las entidades de la Administración Pública previa la instalación del software libre en sus equipos, deberán verificar la existencia de capacidad técnica que brinde el soporte necesario para el uso de este tipo de software.

Artículo 4. Se faculta la utilización de software propietario (no libre) únicamente cuando no exista una solución de Software Libre que supla las necesidades requeridas, o cuando esté en riesgo la seguridad nacional, o cuando el proyecto informático se encuentre en un punto de no retorno.

Para efectos de este decreto se comprende como seguridad nacional, las garantías para la supervivencia de la colectividad y la defensa de patrimonio nacional.

Para efectos de este decreto se entiende por un punto de no retorno, cuando el sistema o proyecto informático se encuentre en cualquiera de estas condiciones:

- a) Sistemas en producción funcionando satisfactoriamente y que en un análisis de costo beneficio muestre que no es razonable ni conveniente una migración a Software Libre.
- b) Proyecto en estado de desarrollo y que un análisis de costo beneficio muestre que no es conveniente modificar el proyecto y utilizar Software Libre.

Periódicamente se evaluarán los sistemas informáticos que utilizan software propietario con la finalidad de migrarlos a Software Libre.

Artículo 5. Tanto para software libre como software propietario, siempre y cuando se satisfagan los requerimientos, se debe preferir las soluciones en este orden:

- a) Nacionales que permitan autonomía y soberanía tecnológica.
- b) Regionales con componente nacional.
- c) Regionales con proveedores nacionales.
- d) Internacionales con componente nacional.
- e) Internacionales con proveedores nacionales.
- f) Internacionales.

Artículo 6. La Subsecretaría de Informática como órgano regulador y ejecutor de las políticas y proyectos informáticos en las entidades del Gobierno Central deberá realizar el control y seguimiento de este Decreto.

Para todas las evaluaciones constantes en este decreto la Subsecretaría de Informática establecerá los parámetros y metodología obligatorias.

Artículo 7. Aburguesarse de la ejecución de este decreto los señores Ministros Coordinadores y el señor Secretario General de la Administración Pública y Comunicación.

Dado en el Palacio Nacional en la ciudad de San Francisco de Quito, Distrito Metropolitano, el día de hoy 10 de abril de 2008.

Rafael Correa Delgado

PRESIDENTE CONSTITUCIONAL DE LA REPÚBLICA

1.6.4 MARCO CONCEPTUAL

1.6.4.1.DESARROLLO DEL TRABAJO

1.6.4.1.1. SOFTWARE LIBRE

1.6.4.1.1.1.Definición

Para definir lo que es software libre es conveniente realizar un análisis de lo que el término significa. Según los gestores de esta filosofía en el campo del software, y usuarios que comparten la misma ideología software es libertad,

Richard Stallman, uno de los principales promotores de este movimiento señala esta libertad como:

Libertad para ejecutar el programa en cualquier sitio, con cualquier propósito y para siempre.

Libertad para estudiarlo y adaptarlo a las necesidades propias.

Esto exige el acceso al código fuente.

Libertad de redistribución, de modo que se permita colaborar con vecinos y amigos.

Libertad para mejorar el programa y publicar las mejoras.

También exige el código fuente.¹

Richard Stallman, con el fin de promover estas libertades crea un tipo de licencia llamada GPL o General PublicLicence, y desarrolla productos bajo este tipo de licencia e invita a los demás a participar del proceso de mejoramiento continuo, generando un pasamiento que se expandió a

¹DAVILA.Manuel.GNU/Linux y el software libre y sus múltiples aplicaciones. 2009

barias comunidades de usuarios los mismos que apoyan esta ideología mediante la creación o modificación de software con este tipo de licencia.

La libertad de ejecutar los programas o aplicaciones significa la libertad para cualquier tipo de persona u organización de usarlo en cualquier tipo de sistema de computación, para cualquier tipo de trabajo y propósito, sin estar obligado a comunicarlo a su programador, o alguna otra entidad específica. En esta libertad, el propósito de los usuarios es el que importa, no el propósito de los programadores. Como usuario es libre de ejecutar un programa para sus propósitos; y si lo distribuye a otra persona, también es libre para ejecutarlo para sus propósitos, pero usted no tiene derecho a imponerle sus propios propósitos.

A menudo en el término software libre es confundido con gratis debido a la traducción que tiene este término, que proviene del idioma inglés (free software). Sin embargo esto no necesariamente tiene que ser así, este software puede ser distribuido con un cierto valor económico, pero sigue conservado las libertades de la licencia.

Para que la libertad 1 y 3 para realizar cambios y publicar versiones mejoradas, tengan sentido; debe tener acceso al código fuente del programa. Por consiguiente, el acceso al código fuente es una condición necesaria para el software libre. El «código fuente» ofuscado no es código fuente real, y no cuenta como código fuente.

La libertad 1 incluye la libertad de usar su versión modificada en lugar de la original. Si el programa se entrega con un producto diseñado para ejecutar versiones modificadas de terceros, pero rechaza ejecutar las suyas, una práctica conocida como «tivoization» o «lockdown o «arranque seguro» (en la terminología perversa de los que la practican); la libertad 1

se convierte más en una ficción teórica que en una libertad práctica. Esto no es suficiente. En otras palabras, estos binarios no son software libre, incluso si se compilaron desde un código fuente que es libre.

Una manera importante de modificar un programa es fusionando subrutinas y módulos libres disponibles. Si la licencia del programa dice que no puede fusionar un módulo existente con una debida licencia, así como si le requiere ser el titular de los derechos de autor de lo que agregue, entonces la licencia es demasiado restrictiva para calificarla como libre.

La libertad 3 incluye la libertad de liberar sus versiones modificadas como software libre. Una licencia también puede permitir otras formas de relicenciarlas, en otras palabras, no tiene que ser una licencia de copyleft. No obstante, una licencia que requiera que las versiones modificadas no sean libres, no se puede considerar como una licencia libre.

El software libre se puede vender si se desea, aunque para penetrar en el mercado es necesario acompañarlo de servicios al usuario; es por esta razón que aparecen las famosas distribuciones de Linux como Debían, Red Hat, Suse" y muchas más, las cuales incluyen el beneficio de llevar a las manos de los usuarios las primeras versiones para su uso. El valor agregado de estas empresas radica en el trabajo que hacen para seleccionar los programas estables, compilarlos en algunos casos, empaquetar el producto final y crear el software adecuado para el manejo de las instalaciones. En casos más especializados se cobra el servicio de soporte y las actualizaciones.

Pero se debe aclarar que un usuario que adquiere este software está en su derecho de regalarlo, venderlo, alquilarlo y hacer cualquier transacción

comercial que sea capaz de llevar a cabo, actividad que el mercado mismo se ha encargado de dificultar debido a que los permisos del software libre de la licencia GPL permiten la copia legal.

La persona que aprenda a acceder a la tecnología de software libre obtendrá grandes beneficios, podrá crear su propio rumbo tecnológico, convertirse en experto y asesorar a los demás.

Existen algunas denominaciones que es importante conocer para hablar con propiedad del tema como es el caso del Open Source Software o Software Abierto, cuyo objetivo es crear estándares abiertos, en principio para una implementación del sistema operacional Unix.

La Open Software Initiative, tiene como objetivo promover el código abierto y es liderada principalmente por Erik Raymond. Hace énfasis en la producción de mejoras al software para crear nuevos productos beneficiando más a los desarrolladores que a los usuarios, en contraposición del software libre de Richard Stallman quien desde la Free Software Foundation promueve la libertad en el uso para beneficio de los usuarios.

Existe también el software denominado Freeware, que es un tipo de licenciamiento que no tiene costo pero ayuda a promover otros desarrollos sin ser libre, más bien abierto, como es el caso del lenguaje Java que no entrega los programas fuentes.

El Shareware, en realidad, es un mecanismo de mercadeo en el cual se dejan usar los programas binarios por un tiempo (con limitaciones), mientras el usuario decide la compra.

Charityware o Careware, Libera los productos pero solicita colaboraciones voluntarias a sus usuarios, como es el caso del software de dominio público, en el cual el autor renuncia de manera explícita a todos sus derechos para bien de los demás.

Copyleft es un caso particular de software libre, que obliga a que las mejoras también sean libres.

Y finalmente, el software propietario, cuyo uso no es libre. Al respecto de esta denominación, Richard Stallman se refiere a él con el nombre software privativo para indicar que priva a los usuarios de las libertades para compartirlo y utilizarlo.

Para efectos de conservar cierta independencia y evitar compromisos con esta discusión, en el libro se utilizará el término "software propietario" para referirse más al concepto de la adquisición de las licencias que al concepto social que promueve Stallman.²

1.6.4.1.1.2.Historia

El software libre como concepto no apareció hasta principios de la década de 1980. Sin embargo, su historia puede trazarse desde bastantes años antes.

Entre los años 1960 y 1970, el software no era considerado un producto sino un añadido que los vendedores de las grandes computadoras de la época (las mainframes) aportaban a sus clientes para que éstos pudieran usarlos. En dicha cultura, era común que los programadores y

²DAVILA.Manuel.GNU/Linux y el software libre y sus múltiples aplicaciones. 2009

desarrolladores de software compartieran libremente sus programas unos con otros.

Durante los años 1960 el panorama de la informática estaba dominado por los grandes ordenadores, instalados fundamentalmente en empresas y centros gubernamentales.

A mediados de la década de 1970 era ya absolutamente habitual, en cualquier ámbito informático, encontrarse con software propietario. Esto supuso un gran cambio cultural entre los profesionales que trabajaban con software, a la vez que un gran número de empresas florecían en torno al nuevo negocio.

Incluso cuando la tendencia abrumadoramente mayoritaria era la de explorar el modelo de software propietario, había iniciativas que mostraban algunas características de lo que luego se consideraría software libre. Algunas de ellas incluso produjeron software libre, según la definición que usamos hoy día. Entre ellas, caben destacar Spiced TeX y TeX, además del caso mucho más complejo de Unix.

Unix fue uno de los primeros sistemas operativos portables, creado originalmente por Thompson y Ritchie (entre otros) en los Bell Labs de AT&T. Su desarrollo ha continuado desde su nacimiento, hacia 1972, hasta hoy, con innumerables variantes comercializadas por literalmente decenas de empresas.

A principios de 1984, Richard Stallman, en aquella época empleado en el AI Lab del MIT, abandonó su trabajo para comenzar el proyecto GNU. Desde el principio del proyecto GNU, Richard Stallman estaba preocupado por las libertades que tendrían los usuarios de su software.

Estaba interesado en que no sólo los que recibieran los programas directamente del proyecto GNU, sino cualquiera que lo recibiera después de cualquier número de redistribuciones y (quizás) modificaciones, siguiera disfrutando de los mismos derechos (modificación, redistribución, etc.). Para ello, escribió la licencia GPL, probablemente la primera licencia de software diseñada específicamente para garantizar que un programa fuera libre en este sentido. Al mecanismo genérico que utilizan las licencias tipo GPL para conseguir estas garantías, Richard Stallman lo llamó copyleft, que hoy día es el nombre de una gran familia de licencias de software libre.

Richard Stallman también fundó la Free Software Foundation (FSF con el fin de conseguir fondos para el desarrollo y la protección del software libre, y sentó los fundamentos éticos del software libre, con documentos como The GNU Manifesto y Why Software Should Not Have Owners.³

1.6.4.1.1.3.Licencias libres

Existen una gran cantidad de licencias para regular el uso del software libre estas licencias son el instrumento jurídico que garantiza legalmente las cuatro libertades del software libre que son (uso, copia, modificación y redistribución), permite identificar a un programa o aplicación es software libre:

En síntesis:

- Las licencias fijan las condiciones de uso y distribución de una obra.
- Sin licencias no existiría la posibilidad de liberar las obras.

³ GONZÁLEZ.J, SEOANE .J. ROBLES. G. Introducción al software libre. 2003

- Las licencias libres son más que una licencia: son una declaración de principios, un contrato social.⁴

1.6.4.1.1.3.1. Tipos de licencias

1.6.4.1.1.3.1.1.Licencias GPL

La Licencia Pública General (inglés: General Public License o GPL).⁵

Este tipo de licencia permite al usuario la libertad de compartir, cambiar o modificar y distribuir el software bajo este tipo de licencia.

1.6.4.1.1.3.1.2.Licencias AGPL

La Licencia Pública General de Affero (en inglés Affero General Public License, también Affero GPL o AGPL).⁶

Es un tipo de licencia derivada de la licencia GPL, la cual obliga a distribuir el código fuente en aplicaciones que se ejecutan en servicios de red.

1.6.4.1.1.3.1.3.La LGPL (Licencia Pública General Reducida de GNU)

Una particularidad de la LGPL es que permite que software bajo sus términos sea convertido en software GPL. Esta característica es útil si se desea reutilizar código de una librería LGPL en aplicaciones o

⁴VIDAL.Miguel. Informe sobre licencias libres.2008

⁵GARCIA.Leonardo.Licencias de software.2007

⁶VIDAL.Miguel.Informe sobre licencias libres.2008

librerías GPL (por ejemplo, con el fin de impedir que sean utilizadas por software privativo).⁷

1.6.4.1.1.3.1.4.Licencias estilo BSD

La Licencia de Distribución de Software de Berkeley (inglés: Berkeley Software Distribution o BSD).

Es un tipo de licencia permite a los programadores la utilizar, modificar, distribuir o compartir el código fuente de un software.⁸

1.6.4.1.1.3.1.5.Licencias estilo MPL y derivadas

La Licencia Pública de Mozilla (inglés: Mozilla PublicLicenseó MPL). Es un tipo de licencia de software libre principalmente utilizada por el navegador de internet Mozilla y sus productos. Está dirigida a una reutilización del software si el usuario desea, si restringir la reutilización del código ni el relicenciamiento bajo la misma licencia. Esta licencia es utilizada por programadores para liberar su código fuente.⁹

1.6.4.1.1.3.1.6.Copyleft

El Copyleft es un grupo de licencias que se refiere al derecho de autor que consiste en la libre utilización, distribución y modificación de un producto, siempre y cuando los derechos libres se mantengan.

⁷VIDAL.Miguel.LGPL: la Licencia P´ublica General Reducida de GNU.2009

⁸GARCIA.Leonardo.Licencias de software.2007

⁹GARCIA.Leonardo.Licencias de software.2007

- Se refiere a un tipo de licencias de software libre (y por lo tanto, a un tipo de software libre): aquellas que permiten la redistribución del software sólo si se garantiza a quien lo recibe libertades iguales a las que otorgó el productor del programa.¹⁰

1.6.4.1.1.4. Ventajas del software libre.

El software libre presenta una serie de ventajas sobre el software propietario por los derechos que otorga a sus usuarios. Algunas de estas ventajas pueden ser más apreciadas por los usuarios particulares, otras por las empresas, y otras por las administraciones públicas.

1. Bajo costo de adquisición y libre uso.

En el caso de software libre, comercialmente no está disponible, lo que se adquiere por un costo bajo o nulo es la licencia con respecto al uso que se puede dar. Esto permite a disminuir el costo de gastos por mantenimiento lo que significa un ahorro económico.

El software libre por su naturaleza el usuario puede adquirir e instalar en las maquinas que necesite.¹¹

2. Innovación tecnológica.

El principal objetivo del software libre es la colaboración, con lo cual crece y se innova las aplicaciones informáticas, sistemas operativos y demás software, esto se logra debido al pensamiento de sus impulsores que se basan en la ideología que el conocimiento pertenece a la humanidad, y no debe ser limitada.

¹⁰VV.AA. Copyleft. Manual de uso. 2006

¹¹ GONZÁLEZ.J, SEOANE .J. ROBLES. G. Introducción al software libre. 2003

3. Requisitos de hardware menores y durabilidad de las soluciones.

El software libre en algunos casos no requiere muchos requisitos en hardware y por lo tanto es más barato de implementar.

4. Escrutinio público.

El software libre se basa en la cooperación de programadores voluntarios en su gran mayoría que trabajan de manera coordinada a través de internet para la creación, diseño y rediseño de software.

5. Independencia del proveedor.

Debido al acceso y disponibilidad del código fuente en el software libre existe independencia con respecto al distribuidor del software.

6. Adaptación del software.

Gracias a la disponibilidad del código fuente el usuario o la organización puede adaptar el software que necesita según sus requerimientos ya que puede agregar o a su vez eliminar partes del software hasta adaptar exactamente a nuestra necesidad. Lo que no sucede con el software propietario que siempre se distribuye paquetes de software debido a un estudio de mercado y no adaptado a las necesidades de empresas o administradores.

1.6.4.1.1.5. Ventajas y desventajas del software libre y propietario

En el siguiente cuadro comparativo se evidencia algunas de las ventajas y desventajas que poseen el software libre y el software propietario.

	Ventajas	Desventajas
Software libre	<p>Seguro y estable</p> <p>Es gratuito</p> <p>No requiere licencia de propietario.</p> <p>Posee código abierto.</p> <p>Es modificable.</p> <p>Posee libertad de copia.</p> <p>Es libre para distribuirlo.</p>	<p>Muy poca compatibilidad con el hardware.</p> <p>Es muy poco utilizado.</p> <p>Se necesita un grado de conocimientos para poder utilizarlos.</p> <p>Dificultad en el intercambio de archivos.</p> <p>No posee control de calidad.</p> <p>Posee una interfaz sencilla y compleja.</p>
Software propietario	<p>Documentación completa.</p> <p>Posee control de calidad.</p> <p>Ocupa la mayor parte del mercado.</p> <p>Existen una gran cantidad de aplicaciones.</p> <p>Posee soporte técnico.</p> <p>Posee una interfaz perceptible y mejor diseñada.</p>	<p>Es inestable.</p> <p>No es muy seguro.</p> <p>Necesita antivirus</p> <p>Tiene costo.</p> <p>Requiere licencias del propietario.</p> <p>Es diseñado para ser usado.</p> <p>Su código fuente es cerrado.</p> <p>Se necesita costos cursos para poder aprender a utilizarlos.</p> <p>Posee restricciones.</p> <p>No es modificable.</p>

Cuadro 1. Ventajas y desventajas del software libre y propietario

Fuente: (Flor Beatriz Arévalo Quinabanda)

1.6.4.1.1.6.La educación y el conocimiento libre

Uno de los denominadores comunes de muchos de los autores cuyos artículos se recogen en esta colección es su estrecha relación con el mundo de la docencia, en particular con la docencia universitaria en el campo de la informática. Por eso, cuentan con una especial sensibilidad para temas relacionados con la educación y la difusión de conocimiento.

El uso de herramientas informáticas libres se está convirtiendo en habitual en institutos y universidades en los últimos años. Esto es debido a varias características del software libre que se tornan en grandes ventajas de cara a la docencia, como se puede leer en Software libre en la enseñanza de la informática, el primero de los ensayos de esta parte. Piénsese, por ejemplo, en el hecho de que los profesores pueden distribuir los programas de forma gratuita a sus alumnos o, muy importante en el caso de estudios informáticos, se puede inspeccionar el código fuente.

La posibilidad de inspeccionar el código fuente y, en general la estructura de los programas, puede parecer un hecho menor, pero sin lugar a dudas influye de manera notable en los planes de estudio de las carreras informáticas. Así, de las aplicaciones que casi todo usuario suele utilizar sistemas operativos, suites de ofimática y recientemente aplicaciones para Internet, solo los sistemas operativos y las aplicaciones para Internet suelen ser estudiadas de manera minuciosa en la universidad; las asignaturas de creación de herramientas ofimáticas brillan por su ausencia.

Entre otras razones, podemos suponer que esto es debido a que hasta hace poco no hemos estado en disposición de conocer cómo se construyen ni de estudiar de cerca casos prácticos de esas herramientas. Ciertamente, esto

está cambiando gracias al software libre y a las nuevas posibilidades que ofrece.

En este sentido, las universidades punteras están empezando a cambiar de rumbo y empiezan a f iniciativas para dar difusión a sus cursos, algo que hasta ahora lo hacían como mucho los profesores a nivel particular. Esto no es otra cosa que una generalización de las ideas que hacen al software libre y aplicadas al conocimiento. En De como el conocimiento puede ser libre se presenta una iniciativa -probablemente la más conocida- por parte de una prestigiosa universidad americana. Si ya ahora es una referencia, a partir de ahora y con iniciativas como esta, lo sería más.

Finalmente, en ¿Que tiene que estudiar un informático? se plantean una serie de ideas que ya han sido expuestas en los dos artículos anteriores, pero esta vez orientadas de manera concreta a las carreras de ingeniería en informática.¹²

1.6.4.1.1.7. Software libre en la enseñanza informática

Hace ya tiempo que, en mayor o menor medida, los ordenadores han entrado en las escuelas, en los institutos, y desde luego en las universidades. En muchos casos con ellos se practican conocimientos específicamente informáticos, pero cada vez más se usan sobre todo como herramienta para enseñar otro tipo de disciplinas, o simplemente para permitir a los alumnos practicar con herramientas genéricas (ofimáticas, de consulta de web, etc.).

La inmensa mayoría de estos ordenadores utiliza software propietario, y en particular alguna versión de Microsoft Windows y Microsoft Office.

¹² GONZÁLEZ. Jesús M. La educación y el conocimiento libre.2002

Sin embargo, la elección de estos programas raramente es una decisión meditada, ni suele estar basada en un análisis de las opciones disponibles. Es más, en muchos casos ni siquiera los responsables de esta decisión son conscientes de que existen otras opciones. Pero estas otras opciones existen, y entre ellas destaca por sus ventajas la basada en software libre.¹³

1.6.4.1.2. APLICACIÓN WEB

1.6.4.1.2.1. Definición

En Ingeniería de software una aplicación Web es una “aplicación a la cual se tiene acceso vía un navegador Web sobre una red, como Internet o una intranet. Además, es una aplicación de software codificada en un lenguaje soportado por un browser o navegador Web (Como HTML, JavaScript, Java), en la que se confía la ejecución al navegador.”

En Ingeniería de software una aplicación Web es una “aplicación a la cual se tiene acceso vía un navegador Web sobre una red, como Internet o una intranet. Además, es una aplicación de software codificada en un lenguaje soportado por un browser o navegador Web (Como HTML, JavaScript, Java), en la que se confía la ejecución al navegador.”¹⁴

Las aplicaciones web permiten la generación automática de contenido, la creación de páginas personalizadas según el perfil del usuario o el desarrollo del comercio electrónico. Además, una aplicación web permite interactuar con los sistemas informáticos de gestión de una empresa, como

¹³ GONZÁLEZ. Jesús M. Software libre en la enseñanza informática.2002

¹⁴ MONTALVO.Enrique,RODRIGUEZ.Miguel.Modelo de seguridad en las aplicaciones web desarrolladas por un tercero.2009.

puede ser gestión de clientes, contabilidad o inventario, a través de una página web.

Las aplicaciones web se encuadran dentro de las arquitecturas cliente/servidor: un ordenador solicita servicios (el cliente) y otro está a la espera de recibir solicitudes y las responde (el servidor).¹⁵

Es importante mencionar que una página Web puede contener elementos que permiten una comunicación activa entre el usuario y la información. Esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responde a cada una de sus acciones, como por ejemplo; rellenar y enviar formularios, participar en juegos diversos y acceder a gestores de base de datos de todo tipo.

1.6.4.1.2.2. Antecedentes de las aplicaciones web

Inicialmente la web era simplemente una colección de páginas estáticas, documentos, etc., que podían consultarse o descargarse.¹⁶

En los primeros programas de computación que utilizaban una arquitectura cliente-servidor, cada aplicación tenía su propio programa cliente que servía como interfaz de usuario que tenía que ser instalado por separado en cada computador personal de cada usuario. El cliente realizaba peticiones a otro programa, -el servidor-, que le daba respuesta. Una mejora en el servidor, como parte de la aplicación, requería normalmente una mejora de los clientes instalados en cada ordenador personal, añadiendo un coste de soporte técnico y disminuyendo la productividad.

¹⁵ MORA. Programación de aplicaciones web: historia, principios básicos y clientes web.

¹⁶ CARLES MATEU. Desarrollo de aplicaciones web. 2004

En contraste, las aplicaciones Web generan dinámicamente una serie de páginas en un formato estándar como HTML o XHTML que son soportados por los navegadores Web comunes. Para añadir más dinamismo a este proceso, se utilizan lenguajes interpretados en el lado del cliente, tales como JavaScript, que agrega elementos dinámicos a la interfaz de usuario.

Generalmente cada página Web en particular se envía al cliente como un documento estático, pero la secuencia de páginas ofrece al usuario una experiencia interactiva. Durante la sesión, el navegador Web interpreta y muestra en pantalla las páginas, actuando como cliente para cualquier aplicación Web.

A lo largo del tiempo, y con la mayor difusión de navegadores Web en las aplicaciones informáticas de la vida cotidiana, estos han sufrido diversas evoluciones, todas ellas, encaminadas a proporcionar una mayor potencia en la interacción del navegador con el usuario, aplicando la inserción de lenguajes de programación de secuenciado de instrucciones, o scripting.

En 1995, Netscape introdujo un programa de scripting, llamado JavaScript, que permitía a los programadores agregar algunos elementos dinámicos a la interfaz de usuario, que corría del lado del cliente. Hasta entonces, todos los datos tenían que ser mandados al servidor para su procesamiento y los resultados eran entregados al cliente a través de páginas HTML estáticas.

En 1996, Macromedia introdujo Flash; un reproductor de animaciones vectoriales que podía ser agregado al navegador Web como un plug-in, o complemento, para incrustar animaciones en las páginas Web. Esto

permitía el uso de lenguaje de scripting para programar interacciones en el lado del cliente sin necesidad de comunicarse con el servidor.

En 1999 el concepto “aplicación Web”, fue introducido al lenguaje Java en lo que fue denominado Servlet 2.2. En ese tiempo, ambos, JavaScript y XML ya habían sido lanzados, pero AJAX, aún no había sido concebido, y los objetos XMLHttpRequest habían sido introducidos en Internet Explorer 5.

En 2005, Ajax fue acuñado, y aplicaciones como Gmail empezaron a hacer el lado cliente más y más interactivo.¹⁷

1.6.4.1.2.3.Consideraciones técnicas

De acuerdo al sitio Ferran Barba, “una ventaja significativa es que las aplicaciones Web deben funcionar igual, independientemente de la versión del sistema operativo instalado en el cliente.”

En vez de crear clientes para Windows, Mac OS X, GNU/Linux, y otros sistemas operativos, la aplicación Web se escribe una vez y se ejecuta igual en todas partes. Sin embargo, hay aplicaciones inconsistentes escritas con HTML, CSS, DOM y otras especificaciones para navegadores Web que pueden causar problemas en el desarrollo y soporte de las aplicaciones Web.

Adicionalmente, la posibilidad de los usuarios de personalizar muchas de las características de la interfaz (tamaño y color de fuentes, tipos de

¹⁷MONTALVO.Enrique,RODRIGUEZ.Miguel.Modelo de seguridad en las aplicaciones web desarrolladas por un tercero.2009.

fuentes, inhabilitar JavaScript) puede interferir con la consistencia de la aplicación Web.

Otra aproximación es utilizar Adobe Flash Player o Java applets para desarrollar parte o toda la interfaz de usuario. Como casi todos los navegadores incluyen soporte para estas tecnologías (usualmente por medio de plug-ins), las aplicaciones basadas en Flash o Java pueden ser implementadas con aproximadamente la misma facilidad. Dado que ignoran las configuraciones de los navegadores, estas tecnologías permiten más control sobre la interfaz, aunque las incompatibilidades entre implementaciones Flash o Java puedan crear nuevas complicaciones.¹⁸

1.6.4.1.2.4. Estructura de las aplicaciones web.

Aunque existen muchas variaciones posibles, “una aplicación Web está normalmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador Web ofrece la primera capa y un motor capaz de usar alguna tecnología Web dinámica (PHP, ASP, ASP.NET, CGI, ColdFusion, embPerl, Python (programming language) o Ruby on Rails); que constituye la capa de en medio. Por último, una base de datos constituye la tercera y última capa.

El navegador Web manda peticiones a la capa de en medio que ofrece servicios valiéndose de consultas y actualizaciones a la base de datos y a su vez proporciona una interfaz de usuario”¹⁹

¹⁸MONTALVO.Enrique,RODRIGUEZ.Miguel.Modelo de seguridad en las aplicaciones web desarrolladas por un tercero.2009.

¹⁹MONTALVO.Enrique,RODRIGUEZ.Miguel.Modelo de seguridad en las aplicaciones web desarrolladas por un tercero.2009.

1.6.4.1.2.5.Lenguajes de programación

Existen numerosos lenguajes de programación empleados para el desarrollo de Aplicaciones Web, entre los que destacan:

- PHP
- ASP/ASP.NET
- JAVA, con sus tecnologías Java Servlets y JavaServerPages (JSP)
- Perl
- HTML
- XML
- Ruby
- Python

De acuerdo a este criterio 20, ASP no es un lenguaje de programación en sí mismo, sino una arquitectura de desarrollo Web en la que se pueden usar por debajo distintos lenguajes (por ejemplo, VB.NET o C# para ASP.NET o VBScript/JScript para ASP).

1.6.4.1.3. USO DE SOFTWARE LIBRE EN APLICACIONES WEB.

Hoy en la actualidad la elección del software y el tipo de plataforma determina una gran inversión, además del problema de las licencias, y la confiabilidad que este software posee. Sin embargo, existe desde hace algunos años una opción para acceder al uso de software, denominada Software Libre.

El Software Libre es aquel que una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente, generalmente

de manera gratuita. El software libre suele estar disponible gratuitamente en Internet, o a precio del costo de la distribución a través de otros medios.

En la actualidad, cualquier aplicación profesional para la web, tiene que hacer uso de conexiones a bases de datos, interpretación de ficheros XML, gestión de plantillas de diseño independientes del código, además de prestar gran atención a la seguridad y poder crear una estructura de desarrollo organizada que permita rápidas actualizaciones y con los mínimos errores, para lo cual el software libre en la actualidad posee grandes alternativas de las cuales son muy usadas. Actualmente está instaladas en millones de computadoras y en compañías como en Google, Amazon, Facebook, Twitter y otras importantes plataformas de la web.

1.6.4.1.4. CARACTERÍSTICAS DEL SOFTWARE LIBRE

Dentro de las características que posee el software libre frente al software propietario tenemos las siguientes:

SOFTWARE LIBRE	SOFTWARE PROPIETARIO
Es desarrollado por una comunidad de voluntarios, y cualquier persona puede participar en el proceso de construcción del software. Esto implica que el código fuente ha de ser abierto para todo el mundo, sin restricciones.	Es desarrollado por una empresa y no difunde sus especificidades. El código es cerrado, nadie puede ver cómo está elaborado el programa.
Podemos distribuir el software libremente entre familiares, amigos, compañeros de trabajo, ya que las licencias libres nos lo permite (la	La licencia nos limita a usar ese software por una sola persona (quien lo adquiere). No se puede redistribuir, a no ser que la licencia lo permita.

más habitual es la GPL), teniendo en cuenta que hemos de seguir la normativa que ésta marca.	
Tenemos la seguridad que miles de voluntarios cooperan en la continua mejora del software, corrigiendo posibles errores y poniendo rápidamente el producto al alcance de todos.	Al no tener acceso al código fuente, no podemos corregir los posibles errores que tenga el programa, ni adaptarlo a nuestras necesidades. Dependemos de las variaciones que la empresa propietaria realice con las nuevas versiones.
Aunque software libre no implica necesariamente que el producto sea gratuito, lo habitual es que podamos obtener los programas descargándolos libremente de internet o a través de distribuciones.	Lo habitual es tener que pagar por adquirir estos programas, y muchos de ellos ya sabemos que no son precisamente baratos y asequibles.

Cuadro 2. Características de Software libre y Propietario.
Fuente: (Flor Beatriz Arévalo Quinabanda)

De estas características podemos identificar cada vez se convierte en una alternativa para realizar todo tipo de trabajos, además se puede notar que en poco tiempo el software libre a ocupado gran parte del mercado esto se debe a la colaboración de miles de usuarios que aportan para que la filosofía de la libertad siga presente en todos los ámbitos del software.

Pero para la mayor parte de usuarios que buscan aplicaciones gratuitas, el software libre cubrirá sus necesidades. En el caso de que se esté dispuesto a pagar el coste de tener más funcionalidades o servicios adicionales, el software propietario es una opción. Esto depende del usuario que necesita o requiere implementar una

determinada herramienta, aunque el software libre sigue teniendo más ventajas que el propietario, lo único sobresaliente del software propietario es el predominio en el mercado lugar que ha ganado gracias al marketing y los años de desarrollo de tecnologías informáticas.

1.6.4.1.5. BENEFICIOS DEL SOFTWARE LIBRE

1.6.4.1.5.1.EN LA ÉTICA

El Software Libre tiene sus bases en una ideología que dice el software no debe tener dueños, es un asunto de libertad: la gente debería ser libre de usarlo en todas las formas que sean socialmente útiles.

De esta forma, el movimiento del Software Libre pone lo que es beneficioso para la sociedad por encima de los intereses económicos o políticos.

Entre los beneficios que percibe la sociedad podemos mencionar:

- Tecnologías transparentes, confiables y seguras.
- Tecnologías como bien público.
- Promoción del espíritu cooperativo, en el que el principal objetivo es ayudar a su vecino.
- Precios justos.

1.6.4.1.5.2.EN LA PRÁCTICA

El Software Libre ofrece a las personas la posibilidad de utilizar, estudiar, modificar, copiar y redistribuir el software. Para hacer efectivas estas libertades, el código fuente de los programas debe estar disponible.

Gracias a estas libertades obtenemos muchos beneficios prácticos:

- Podemos ejecutar el software cuando queramos y para lo que queramos.
- Podemos aprender de los programas existentes.
- Podemos mejorarlos.
- Podemos adaptarlos para que se ajusten a nuestras necesidades.
- Podemos basarnos en ellos, de forma que evitamos los costos adicionales de empezar un programa desde 0.
- Podemos formar negocios alrededor de la creación, distribución, soporte y capacitación de programas libres.

Y el efecto de todos estos beneficios es la formación de Comunidades enormes alrededor de proyectos de software libre, gracias a las cuales tenemos acceso a desarrolladores, documentadores y testers de todo el mundo.

1.6.4.1.6. MOTIVACIÓN PARA EL USO DEL SOFTWARE LIBRE

La motivación ética, abanderada por la Free Software Foundation, heredera de la cultura hacker, y partidaria del apelativo libre, que argumenta que el software es conocimiento y debe poderse difundir sin trabas. Su ocultación es una actitud antisocial y la posibilidad de modificar programas es una forma de libertad de expresión, aunque sin olvidar una estructura jerarquizada por la meritocracia.²⁰

²⁰ Eric S. Raymond.(1997)*La Catedral y el Bazar*.

La motivación pragmática, abanderada por la Open Source Initiative y partidaria del apelativo abierto, que argumenta ventajas técnicas y económicas, con respecto a evitar una tragedia de los anticomunes mejorando los incentivos.²¹

Aparte de estas dos grandes motivaciones, la gente que trabaja en software libre suele hacerlo por muchas otras razones, que van desde la diversión a la mera retribución económica, que es posible debido a modelos de negocio sustentables.

1.6.4.1.7. SOFTWARE LIBRE PARA DESARROLLO DE PÁGINAS WEB

En el pasado la creación de páginas web resultaba una tarea que se realizaba de acuerdo al sistema operativo predominante en el mercado, por esta perspectiva las universidades del país solo se dedicaban a enseñar para el desarrollo web programas de software propietario como ejemplo tenemos Dreamweaver, el software libre era muy poco utilizado ya solo lo utilizaban expertos, o no existía programas que equipare las características de las aplicaciones propietarias, pero en la actualidad, gracias a la colaboración y desarrollo de las tecnologías y nuevas aplicaciones de tipo libre es cada vez una realidad más para los usuarios básicos de la red de redes contar con programas con interfaz más fáciles y dirigidas a usuarios promedios.

Para justificar el uso de programas libres en el desarrollo de páginas web, podemos decir que la Internet misma se fundamenta en los estándares abiertos que fueron desarrollados por estas aplicaciones. Es necesario indicar que el uso de software libre es muy frecuente dentro de la red a si

²¹ Gonzáles Barahona Jesús, Seoane Pascual Joaquín, Robles Gregorio (2003). Introducción al software libre. Barcelona: Eureka media.

por ejemplo tenemos que el servidor de páginas web más utilizado en la Internet es Apache, el 60% de los servidores en Internet lo utilizan, según la estadística mensual publicada en Netcraft. (<http://news.netcraft.com/>)

En dicho sitio se ha publicado un conteo que señala que la tercera parte de los sitios activos en Internet están soportados en Linux.

El software libre en la actualidad ofrece algunas alternativas gratuitas con herramientas que facilitan la creación de páginas web, por lo que se convierte en una interesante opción para diseñar sitios web.

1.7 COMPROBACIÓN DE LA HIPÓTESIS

Al inicio del presente trabajo investigativo se planteó la siguiente hipótesis: EL USO DE SOFTWARE LIBRE PERMITIRÁ UNA FORMACIÓN PROFESIONAL E INTEGRAL EN EL DESARROLLO DE APLICACIONES WEB EN LOS ESTUDIANTES DE LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS.

Para evidenciar lo antes referido se aplicó un sistema de estadística descriptiva apoyada en el método porcentual descrito en los cuadros y gráfica.

Para comprobar lo manifestado acudimos al uso de la estadística descriptiva apoyados por el método porcentual detallado en los cuadros, gráficos y su respectiva interpretación.

1.8 METODOLÓGIA

1.8.1 MÉTODOS

Método Analítico: Se analizó el desarrollo de la aplicación web ya que su proceso se realiza por partes estudiando en la carrera de Tecnología en Informática Aplicada.

Método Deductivo: Los conceptos que utilizados para efectuar el estudio de las aplicaciones web, y extraer los principios del software libre, que servirán para conclusiones en las cuales se aplican, parte de las verdades previamente establecidas como principio general luego aplicarlo a casos individuales y comprobar así su validez.

Método Descriptivo: Se observó en el desarrollo de aplicaciones web, fenómenos, casos y no se limitó a la simple recolección y tabulación de datos, sino que se procurara la interpretación racional y el análisis del objetivo de los mismos con la finalidad que se estableció previamente.

Método Inductivo: Método utilizado para hacer un proceso de análisis y síntesis del desarrollo de aplicaciones web.

1.9 TIPO DE INVESTIGACIÓN

La presente investigación tiene el siguiente diseño:

Por el propósito es Aplicada.- Ya se realizó con fines prácticos, tanto para resolver un problema, como para tomar decisiones, evaluar programas, y en

general, para mejorar un producto o proceso por medio del estudio de conceptos teóricos en situaciones reales.

Por el nivel es Descriptiva.- Esta investigación es de tipo descriptiva ya que por medio de la misma describiremos los hechos y fenómenos que intervienen en la investigación.

Por el lugar es de Campo.- Este trabajo tiene un diseño de campo, es decir, aplicamos los instrumentos de investigación directamente en el lugar de los hechos, en este caso en la Universidad Estatal de Bolívar.

1.10 TÉCNICAS E INSTRUMENTOS PARA LA OBTENCIÓN DE DATOS

Para la recolección e interpretación de los datos obtenidos se utilizó:

Técnicas

La encuesta.- La encuesta dirigida a los estudiantes y docentes de la carrera de Tecnología en Informática Aplicada de la Escuela de Sistemas en el año 2011-2012.

Instrumentos:

Cuestionario.- Con preguntas de tipo cerradas, para la obtención de datos que permitieron analizar la problemática existente, de forma clara y directa.

1.11 UNIVERSO Y MUESTRA

Se trabajó con el universo total de 12 estudiantes de la Carrera de Tecnología en Informática Aplicada que reciben la asignatura de Diseño de Páginas Web y 1 docente que imparte dicha cátedra.

1.12 PROCESAMIENTO DE DATOS

Para el análisis y procesar los datos se procederá de la siguiente manera:

Cuadros de tabulación con sus respectivos gráficos, resumen, procesamiento de gráficos utilizando OpenOffice.org en programas de Gnumeric y el análisis respectivo de cada pregunta en AbiWord

CAPÍTULO II

2 MARCO PROPOSITIVO

2.1 USO DE SOFTWARE LIBRE PARA EL DESARROLLO DE APLICACIONES WEB EN LA CARRERA DE TECNOLOGÍA EN INFORMÁTICA APLICADA DE LA ESCUELA DE SISTEMAS.

Este documento fue elaborado por la necesidad de fomentar la implementación de software libre en las diferentes áreas de estudio en la Universidad Estatal de Bolívar, de manera especial en la materia de desarrollo de aplicaciones web carrera de Tecnología en Informática Aplicada de la Escuela de Sistemas ya que las tecnologías de información y comunicación son de vital importancia y utilidad en nuestro mundo actual, sin embargo en el área de desarrollo web, implican costos elevados que muchas veces, no retornan la inversión a mediano plazo. Para el caso de empresas y gobiernos locales significa destinar un alto porcentaje de fondos públicos que se requieren para obras y servicios a las comunidades, al pago de licencias de software lo que supone una traba al momento de ofertar un servicio para el desarrollo de proyectos web en estos organismos.

El Software Libre permite que empresas, instituciones públicas y público en general elijan dentro de la amplia variedad de herramientas, aquellas que más se adapten a las necesidades y funciones de la organización o su vida cotidiana. Además, los bajos costos de licencias (o inexistentes) por adquisición del software les permiten centrarse más en los costos de hardware (compra de equipo por ejemplo), capacitación y los servicios que ofrecerán.

Para la Universidad Estatal de Bolívar supone una gran ventaja la utilización del software libre, puesto que el costo que significa el comprar licencias para distribuir a las diferentes extensiones, es alto. La universidad cuenta con laboratorios de cómputo destinados a la investigación pero no podemos hablar

de investigación si en la misma el software no posee las licencias correspondientes, los estudiantes no podrán ofertar sus proyectos realizados en estos laboratorios por el factor de leyes y derechos.

En la actualidad esta realidad está por cambiar ya que se han ejecutado planes de estudio en el que está incluido el software libre, como ejemplo tenemos los estudios en los institutos de informática con programas como Open Office, Libre Office, y el sistema más conocido de este tipo de software como es Ubuntu. Pero en cambio en la escuela de sistemas tecnología en informática y específicamente en desarrollo web no se ha implantado todavía razón por lo que he visto en la necesidad de proponer que se desarrollen planes de estudio para el pensum académico de esta carrera se incluya, en desarrollo web programas de software libre.

2.2 PLAN OPERATIVO

2.2.1 PLAN DE UTILIZACIÓN DE SOFTWARE LIBRE PARA DESARROLLO DE APLICACIONES WEB

En este capítulo se presenta el plan de utilización de software libre para el desarrollo de aplicaciones web en la Universidad Estatal de Bolívar.

2.2.1.1 Objetivo

El objetivo del plan de es reemplazar por software libre todo el software privativo que se utiliza para el desarrollo web en la carrera de tecnología en informática, de tal forma que se puedan realizar las mismas tareas que se llevan a cabo actualmente con software privativo.

2.2.1.2 Consideraciones

En el caso de optar por un cambio, la Universidad podría recurrir a sus propios estudiantes para que aporten ideas y que colaboren en el proceso. Sería una actividad de gran valor agregado para el estudiante poder participar de un emprendimiento tan importante y a su vez complementar los conocimientos teóricos incorporados en las diversas materias.

Para evitar o reducir la resistencia al cambio por parte de los usuarios, la migración debe ser gradual, reemplazando un programa a la vez.

Muchas veces un programa privativo no posee una alternativa equivalente en software libre, pero es posible utilizar varios programas libres que en su conjunto son equivalentes a dicho programa privativo.

2.2.1.3 Fases del proceso de migración a software libre.

El proceso de migración a software libre está compuesto por varias fases, algunas de las cuales se pueden ejecutar en paralelo, mientras que otras deben llevarse a cabo de forma secuencial.

A continuación se explica cada fase del proceso de migración. Se presentan según el orden sugerido de ejecución.

2.2.1.4 Convencer sobre la necesidad de migrar a software libre

La educación relacionada con la informática es hoy día un monocultivo de algunas marcas de software propietario. Sin realizar en muchos casos ningún estudio previo, se elige como plataforma para la formación la que se percibe como la más habitual.

Muchas veces no se tiene en cuenta si esta es la mejor opción posible. O peor aún, se suele confundir la introducción a la informática con un curso de introducción a cierto sistema operativo o los conocimientos sobre ofimática con el conocimiento de una cierta marca de programa ofimático. En general, mucha gente supone que saber de informática es lo mismo que saber manejar ciertas herramientas propietarias.

Por lo general, al justificar la elección de este tipo de educación se indica lo siguiente:

- Es mejor enseñar el uso de la plataforma dominante en el mercado, porque así lo enseñado será más útil al alumno.
- Los propios alumnos piden que se les enseñe el uso de ciertos programas, y piensan que si se usan otros, los conocimientos les van a ser de menos utilidad.
- No hay muchas alternativas, y en cualquier caso, no las hay con ventajas claras sobre el uso de la plataforma dominante.

Estas razones no son válidas. Por otro lado, no es el objetivo de este planteo rechazar la enseñanza basada en herramientas propietarias. El punto de esta afirmación es que al alumno se le debe mostrar que hay alternativas, y que no solamente existen software propietario. Porque en el caso que algún día cambie el dominador del mercado; como ya está ocurriendo ese futuro profesional va a sentirse excluida y no capacitado.

Sería de mayor utilidad enseñarle cómo funciona un editor de código HTML, y no únicamente los detalles del uso de un determinado programa (Dreamweaver) en particular. Naturalmente habrá que hacer unas prácticas, y en ellas lo ideal sería utilizar más de una herramienta.

El uso de software libre debería ser establecido por la Universidad como una política basada en aspectos éticos y de libertad, pues sólo de esa forma el proceso de migración tendrá la fuerza suficiente para enfrentar a los obstáculos con los que se encuentre. Si se utilizan solamente criterios económicos y técnicos, el proceso de migración podría verse amenazado ante la presencia de software privativo que sea superior económica y técnicamente, pero inferior en términos de libertad.

También es importante que la comunidad universitaria en general esté informada de las razones del cambio, por una parte para reducir la resistencia, pero principalmente para que esté consciente de sus libertades en el mundo digital.

Soporte técnico

Se debe evaluar las distintas alternativas libres y decidir cuáles se utilizarán, definir los programas para el estudio de desarrollo web, buscar a los docentes adecuados para impartir estos conocimientos.




2.2.1.5 Identificar programas utilizados actualmente.


En esta etapa se identificó los programas libres y privativos que se utilizan en la Universidad.

Esto se logró mediante reuniones con grupos de usuarios. Los resultados de esta fase permitieron identificar posibles problemas, como software privativo que no tenga una alternativa libre de buena calidad. Además, se pudo saber si actualmente se está utilizando software libre en la Universidad, lo cual facilitaría el proceso de migración.

Se realizó una encuesta a estudiantes y docentes de la carrera de tecnología en informática aplicada de la escuela de sistemas. El objetivo de esta encuesta fue determinar la utilización de software libre para el desarrollo de aplicaciones web, continuación se interpretan los resultados obtenidos:

- Los estudiantes poseen escasos conocimientos en software libre.
- El software libre no es muy utilizado en el estudio por parte de los estudiantes.
- En cátedra de desarrollo web se utiliza solo software privativo.
- En el caso del estudio de desarrollo de aplicaciones web se utiliza en mayor parte software privado, con programas como:

NOMBRE DE LA APLICACIÓN	FUNCIÓN	SISTEMA OPERATIVO	TIPO DE LICENCIA
Adobe Dreamweaver 	<i>Construcción, diseño y edición de sitios y aplicaciones Web</i>	<i>Microsoft Windows, Mac</i>	<i>Software Comercial (Software propietario;)</i>
Xara 3D 	<i>Producción de animaciones 3D</i>	<i>Microsoft Windows</i>	<i>Software Comercial (Software propietario;)</i>
Adobe Flash Professional 	<i>Creación y manipulación de gráficos vectoriales</i>	<i>Microsoft Windows, Mac</i>	<i>Adobe EULA (Software propietario)</i>
Adobe Photoshop	<i>Edición, retoque fotográfico y</i>	<i>Microsoft Windows, Mac</i>	<i>Adobe CLUF (Software</i>

	<p><i>pintura a base de imágenes de mapa de bits (o gráficos rasterizados)</i></p>		<p><i>propietario)</i></p>
---	--	--	----------------------------

Cuadro 3 Programas privativos utilizados para impartir la cátedra de desarrollo web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

2.2.1.6 Alternativas libres

Existen una gran cantidad de programas de código abierto, que pueden ser utilizados para el desarrollo web estos pueden ser ubicados en internet. En la web existen listados de software libre que pueden ser utilizados para buscar programas de este tipo que realicen una determinada función o que reemplacen a un determinado programa privativo.

Como propuesta se sugieren que además de los programas privativos como Dreamweaver, se pueden utilizarse cualquiera de estas aplicaciones que poseen igual o mejores herramientas para el desarrollo web.



2.2.1.7 Tipos de software libre para diseño web



Aplicaciones web

Entre los creadores de aplicaciones web más conocidos y utilizados en software libre son los siguientes:

Editores Web

Entre las aplicaciones que permiten editar las páginas web tenemos los siguientes:

NOMBRE DE LA APLICACIÓN	DESCRIPCIÓN	CARACTERÍSTICAS
 <p>BLUEFISH</p>	<p>Bluefish es un software editor HTML multiplataforma y con licencia GPL, lo que le convierte en software libre.</p> <p>Bluefish está dirigido a diseñadores web experimentados y programadores y se enfoca en la edición de páginas dinámicas e interactivas. Es capaz de reconocer diversos lenguajes de programación y de marcas.</p>	<ul style="list-style-type: none"> • Es una aplicación muy rápida. • Abre múltiples archivos simultáneamente. • Soporta la incorporación de varios proyectos. • Soporta lenguajes como: (HTML, PHP, Javascript, JSP, SQL, XML, Python, Perl, CSS, ColdFusion, Pascal, R, Octave/Matlab). • Posee traducción completa a veintidós idiomas entre los cuales está el español.
 <p>AMAYA</p>	<p>Amaya es una herramienta combinada, compuesta por un navegador web y una herramienta de autor.</p>	<ul style="list-style-type: none"> • Crea, diseña y edita páginas web en HTML • Visualización y edición de dibujos vectoriales. • Multiplataforma, para

	<p>Cualquier página web que se abra puede ser editada inmediatamente. Se pueden ver y generar páginas HTML y XHTML.</p>	<p>Windows, Linux, Mac OS X y más</p>
<p>KOMPOZER</p> 	<p>KompoZer es un editor de páginas web, su objetivo es facilitar el desarrollo de páginas web, gracias a las distintas formas de visualización disponibles en su interfaz.</p>	<p>Está disponible para diversas plataformas: Windows, Mac OS, y Linux.</p> <p>Gestor de sitios FTP (<i>Protocolo de Transferencia de Archivos</i>)</p> <p>Pestañas</p> <p>Editor de CSS (<i>Hojas de Estilo en Cascada</i>)</p> <p>Barras de herramientas personalizables</p>
<p>DRUPAL</p> 	<p>Drupal es un software libre, escrito en PHP, que cuenta con una amplia y activa comunidad de usuarios y desarrolladores que colaboran conjuntamente en su mejora y ampliación.</p>	<p>Esta desarrollado para crear un blog personal o profesional, un portal corporativo, una tienda virtual, una red social o comunidad virtual.</p>






Cuadro 4 Aplicaciones web
Fuente: (Flor Beatriz Arévalo Quinabanda)

CAPITULO III






3 ANÁLISIS E INTERPRETACIÓN DE RESULTADOS

3.1 Comparaciones entre lenguajes para el desarrollo de Aplicaciones Web.

Para evaluar la pertinencia de usar un determinado lenguaje de software libre para el desarrollo web se ha seleccionado las siguientes características:

Referencias					
Última versión	5.4.5 (19 de julio de 2012)	1.9.3 (31 de octubre de 2011)	3.2.3 / 2.7.3 (11 de abril de 2012)	Java Standard Edition 7 (1.7.6) (14 de agosto de 2012)	Perl 5, Perl 6
Propósito	Está diseñado para producir páginas web dinámicas	Está diseñado para hacer la programación divertida y flexible.	Está diseñado para enfatizar la productividad y la legibilidad del código.	Creado para funcionar en redes computacionales heterogéneas, e independiente de la plataforma.	Diseñado para varios propósitos, originalmente para la manipulación de texto
Influenciado por:	C, C++, Perl, Java, Python	Perl, Smalltalk, Python, Lisp, Eiffel, Scheme, Ada, Dylan, CLU	ABC, ALGOL 68, C, Haskell, Icon, Lisp, Modula-3, Perl, Smalltalk, Java	Objective-C, C++, Smalltalk, Eiffel	AWK, Smalltalk 80, Lisp, C, C++, Pascal, sed, Unix shell

Sitios que utilizan el lenguaje	Facebook Wikipedia Yahoo Myspace Flickr	Twitter NASA Cisco Amazon BBC	Google Linux Weekly News Youtube Reddit		Terra Videomóvil Use.perl.org
Usabilidad	5,524%	1,691%	3,876%	16,352%	2,273%

TABULACION, PRESENTACION E INTERPRETACION DE RESULTADOS					
Características	PHP 	RUBY 	PYTHON 	JAVA 	PERL 
Interactivo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Multiplataforma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Programación orientada a objetos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Seguridad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sencillo de Aprender	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Simple	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sintaxis clara	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Software libre	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Total	7	7	8	8	6

Cuadro 1 Comparaciones entre los lenguajes orientados a la programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

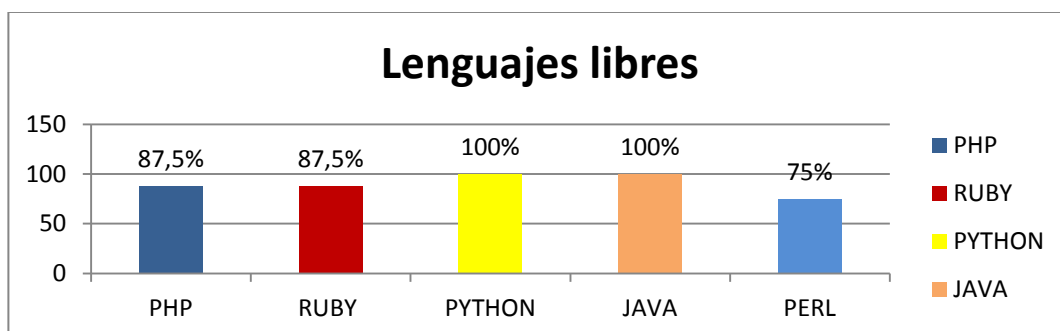


Gráfico 1 Grafica comparativa entre los lenguajes de programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

ANÁLISIS

Encontramos varios lenguajes dentro del software libre que ofrecen herramientas para la construcción de aplicaciones y sitios web. Para identificar un lenguaje idóneo que se utilice para el diseño web, se evaluaron las características de los lenguajes más utilizados en el desarrollo web, a saber: PHP, Ruby, Python, Java y Perl. De acuerdo a las características de cada uno de estos lenguajes como era de esperar, tiene sus pros y sus contras. Pero del análisis de la información recabada se pudo identificar que; PHP es el lenguaje más popular en el mercado ubicándose como uno de los programas líderes, en este campo, gracias a sus ventajas y características, siendo además más demandado en ofertas de trabajo y más conocido por los demandantes de empleo.



Ruby es más usable evitando confusiones en su sintaxis y obteniendo un código más elegante y expresivo, pero más complejo de aprender. Por el contrario, Python por su facilidad de aprendizaje es apropiado para ser el primer lenguaje o para usuarios novatos, y su velocidad de ejecución, es muy aceptable se ejecuta más rápido con diferencia de los demás lenguajes.

De Perl podemos identificar que es un lenguaje muy poco utilizado y conocido en el mercado, pero que a más del resto de lenguajes se pueden identificar algunas características aceptables que posee y pueden ser adaptadas según donde se requiera.

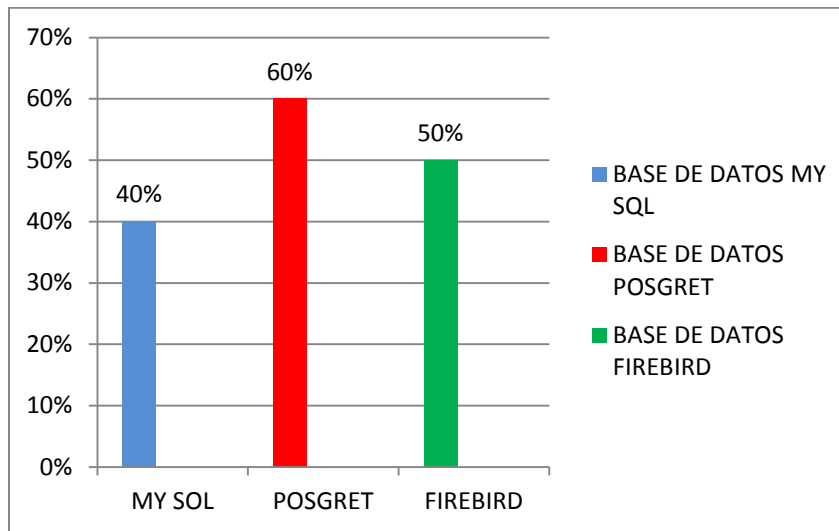
Java es uno de los lenguajes más utilizados a nivel mundial, en la actualidad en cada página web que se puede encontrar en la red de internet, contiene un elemento de este lenguaje, posee una sintaxis sencilla de aprendizaje, clara, interactiva, y orientada a objetos que se ejecuta en varias plataformas de software, por lo que al mirar sus

características podemos identificar como el lenguaje idóneo para introducirse en el mundo del diseño web.

3.2 Comparaciones entre Bases de Datos para el desarrollo de Aplicaciones Web.

Características	MY SQL 	POSGRET 	FIREBIRD 
Claves foráneas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Disparadores	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Restricciones de integridad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alta concurrencia	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Multiplataforma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Seguridad	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Amplia variedad de tipos nativos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Total	4	6	5

Cuadro 2 Comparaciones entre Bases de datos orientados a la programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)



**Gráfico 2 Grafica comparativa entre Base de datos de programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)**

ANÁLISIS

Encontramos vario Base de Datos dentro software libre que ofrecen, herramientas para la construcción de aplicaciones y sitios web. Para identificar una base de datos idóneo que se utilice para diseño web, se a evaluado las características de las bases de datos más utilizados en el desarrollo web a si tenemos MY SQL, POSTGRET, FIREBIRD.

De acuerdo a las características cada uno de las base de datos tienen sus pro y sus contras. Pero del análisis de la información recabada se pudo identificar qué; POSTGRET es una base de datos es más popular en el mercado ubicándose como uno de los programas líderes, en este campo, gracias a sus ventajas y características.

3.3 Comparaciones entre Editores para el desarrollo de Aplicaciones Web.

	KOMPOZER 	BLUEFISH 	AMAYA 
Características			
Abre múltiples documentos	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Barras de herramientas personalizables	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Corrector ortográfico automatizada	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Ligero	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Líneas numeradas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Multiplataforma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Crea, diseña y edita páginas web en HTML	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Edición de fórmulas matemáticas con MathML	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Total	5	6	8

Cuadro 3 Comparaciones entre Editores orientados a la programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

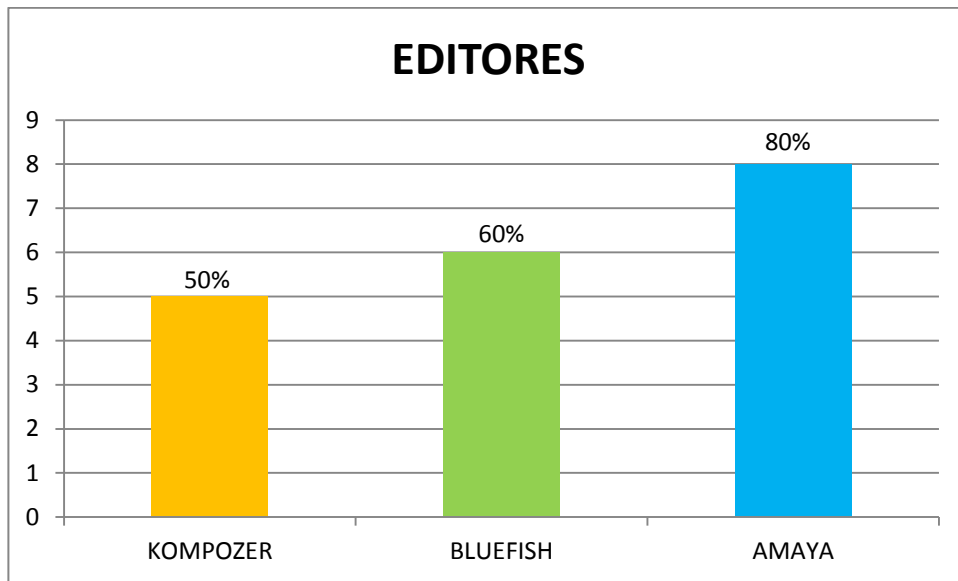




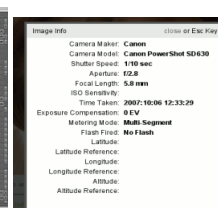
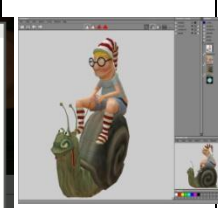
Gráfico 3 Grafica comparativa entre Editores de programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

ANÁLISIS

Encontramos varios Editores dentro del software libre que ofrecen herramientas para la construcción de aplicaciones y sitios web. Para identificar los editores idóneos que se utilicen para el diseño web, se evaluó las características de los editores más utilizados en el desarrollo web, a saber: KOMPOZER, BLUEFISH, AMAYA.

De acuerdo a las características de cada uno de los editores tienen sus pros y sus contras. Pero del análisis de la información recabada se pudo identificar que; AMAYA es un editor más popular en el mercado ubicándose como uno de los programas líderes, en este campo, gracias a sus ventajas y características.

3.4 Comparaciones entre Graficadores para el desarrollo de Aplicaciones Web.

Características	BLENDER	MYPAIN	ZENPHOTO	WINGS 3D
				
Multiplataforma	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Libre	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Modificadores apilables	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Capacidad para hacer Match moving	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Lienzo ilimitado	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Soporte con Google Maps	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Capacidad de bajar todo el álbum como archivo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Total	5	7	4	7

Cuadro 4 Comparaciones entre Graficadores orientados a la programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

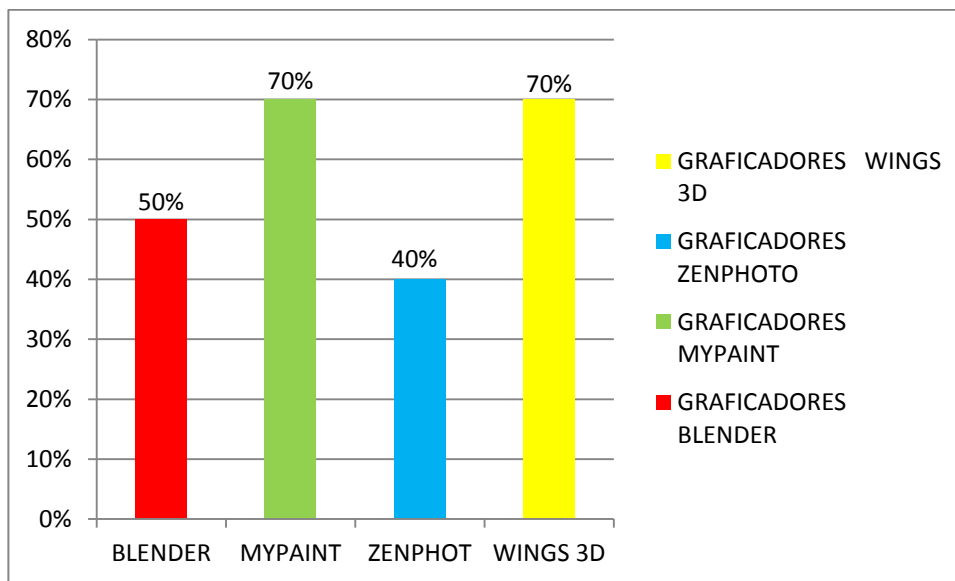


Gráfico 4 Grafica comparativa entre Graficadores de programación web.

Fuente: (Flor Beatriz Arévalo Quinabanda)

ANÁLISIS

Encontramos vario Graficadores dentro del software libre que ofrecen, herramientas para la construcción de aplicaciones y sitios web. Para identificar los Graficadores idóneo que se utilice para diseño web, se a evaluado las características de los Graficadores más utilizados en el desarrollo web a si tenemos: BLENDER, MYPAIN, ZENPHOTO, WINGS 3DDe acuerdo a las características de cada uno de los graficadores tienen su pro y sus contras. Pero del análisis de la información recabada se pudo identificar qué; MYPAIN, WINGS 3D es un de los graficadores más populares en el mercado ubicándose como uno de los programas líderes, en este campo, gracias a sus ventajas y características.

3.5 Comparaciones entre audios para el desarrollo de Aplicaciones Web.

CARACTERISTICAS	AUDACITY	ARDOUR	JOKOSHER
Grabación de audio en tiempo real.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Conversión entre formatos de tipo audio.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Importación de archivos de formato <u>MIDI</u> , <u>RAW</u> y <u>MP3</u> .	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Edición de pistas múltiples.	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Soporte de formatos de audio estándar: wav, wav64, caf, aiff.	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Soporte de audio "mono" y "estéreo"	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
La Facilidad de uso	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Edición y mezcla	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Total	6	3	5

Cuadro 5 Comparaciones entre audios orientados a la programación web.

Fuente: (Flor Beatriz Arévalo Quinabanda)

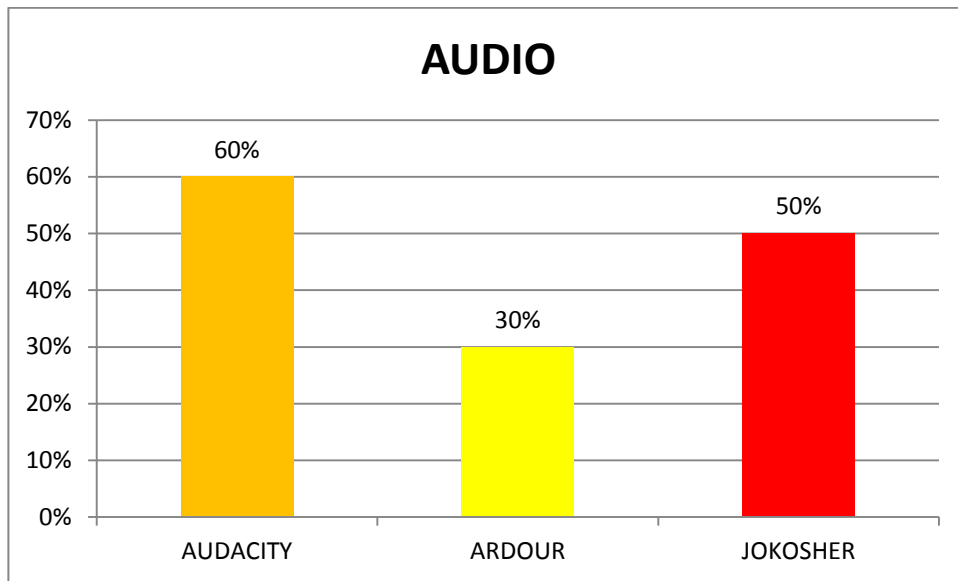


Gráfico 5 Grafica comparativa entre audios de programación web.
Fuente: (Flor Beatriz Arévalo Quinabanda)

ANÁLISIS

Encontramos varios AUDIOS dentro del software libre que ofrecen, herramientas para la construcción de aplicaciones y sitios web. Para identificar los Audios idóneo que se utilice para diseño web, se a evaluado las características de los Audios más utilizados en el desarrollo web a si tenemos: AUDACITY, ARDOUR, JOKOSHER. De acuerdo a las características de cada uno de los Audios tienen su pro y sus contras. Pero del análisis de la información recabada se pudo identificar qué; AUDACITY es un de los Audios más populares en el mercado ubicándose como uno de los programas líderes, en este campo, gracias a sus ventajas y características.

ANÁLISIS FINAL

De acuerdo al análisis en la investigación de diversos programas de software libre pudimos encontrar programas con características mejores que las otras entre ellas podemos hablar de un programas de lenguajes que es el PHP es el lenguaje más popular en el mercado. También podemos hablar bases de datos que es el POSTGRESQL es una base de datos más popular en el mercado, el AMAYA es un editor más popular en el mercado, el MYPAIN, WINGS 3D son los gráficos más populares en el mercado. Se establece que se utilice estas herramientas. Que facilita la construcción de desarrollar aplicación web.

3.6 CONCLUSIONES Y RECOMENDACIONES

3.6.1 CONCLUSIONES:

Una vez concluida la etapa de análisis e interpretación de datos de la información recopilada en el proceso de investigación, determinamos las siguientes conclusiones:

- ❖ Existe una gran cantidad de programas y herramientas que facilitan el desarrollo de aplicaciones web mediante la utilización de software libre, su uso es cada vez más amplio en el mercado llegando a elaborarse proyectos para grandes empresas que existen en la web.

- ❖ Software libre a más de los beneficios económicos que en si implica usar este tipo de tecnologías, permite crear comunidades que mejoran los productos existentes hoy en la web, y que pueden estar disponibles

libremente para su uso. A demás la mayor parte de las herramientas que necesita un usuario final para su trabajo ofimático, o de diseño gráfico, o de programación y desarrollo de aplicaciones existen dentro del ámbito de software libre, y en gran parte de las ocasiones son tan buenas o mejores que las herramientas propietarias.

- ❖ El software libre es una herramienta fundamental para la investigación y el desarrollo ya que por su gran característica de ser de libre acceso y distribución, por lo cual su importancia para su estudio e incorporación en proyectos educativos que busquen fomentar la creación o uso de nuevas ciencias y tecnologías y al realizar un estudio a profundidad sobre este modelo, se considera que la Universidad se vería fuertemente beneficiada de optar por este tipo de software.

3.6.2 RECOMENDACIONES:

Al recolectar la información sobre el presente tema se ha llegado a las siguientes recomendaciones:

- ❖ El software libre ha incursionado en todas las áreas de la informática gracias a la colaboración de miles de usuarios a nivel mundial. En uno de los campos que es más notoria su presencia es en internet, la mayor parte de servidores, sitios o, páginas web están bajo esta política por lo que es necesario implementar en la Universidad Estatal de Bolívar el estudio del software libre en las asignaturas que tienen relación con el diseño web.
- ❖ Se debe promover el uso de software libre en la Universidad Estatal de Bolívar, con énfasis a que los estudiantes se motiven al uso de aplicaciones de tipo libre y así contribuir a dar solución a uno de los problemas en la informática en nuestra localidad como es la utilización de

software de baja calidad o software piratas, por los costes que representa estos programas para comprar originales ya que son de software propietario y esto contribuye a la proliferación de virus informáticos y al daño del hardware de los ordenadores.

- ❖ Es necesario formar profesionales capacitados en el desarrollo de software y no específicamente a usuarios de software propietario en las que sus programas solo se usan y no permiten modificar o arreglar alguna anomalía encontrada y que no hay acceso a su código fuente, evitando así la investigación el estudio y desarrollo.

BIBLIOGRAFÍA

- Guía Práctica sobre Software Libre – Unesco, 2007.
- GUILLERMOPrieto. Metodología para el desarrollo de aplicaciones en la plataforma: Linux PDA, Universidad Anáhuac, México. Enero de 2005.
- INSTITUTO TECNOLÓGICO HOTELERO. Glosario de términos y expresiones de Internet y la Web 2.0.
- Ministerio de Ciencia y Tecnología, noviembre 2004.
- Orden de 21 de febrero de 2005 de la Junta de Andalucía (España), sobre disponibilidad pública de los programas informáticos de la Administración de la Junta de Andalucía y de sus Organismos Autónomos.
- ROXANA.Bassi.diccionario de términos de Internet. Versión 0.6 al 11 de diciembre de 2001.
- SUBSECRETARIA DE INFORMÁTICA. Estrategia Implantación de Software Libre en la Administración Pública Central de Ecuador. Quito, 2009

WEBGRAFIA

- Benjamin Segura, Software libre VS software propietario.
<http://www.portalprogramas.com/milbits/informatica/software-libre-software-propietario.html>
- CulebroJúarez, Montserrat, Gómez Herrera, Wendy Guadalupe, Torres Sánchez, Susana, Software libre vs software propietario Ventajas y desventajas. www.rebellion.org/docs/32693.pdf
- Harris , Brandon. Software libre
- http://es.wikipedia.org/wiki/Software_libre
- <http://www.monografias.com/trabajos12/elsoflib/elsoflib.shtml>
- Kon, MarcosDavid. Software libre.

- López Gascón ,Sergio. Aplicaciones web xml/xslt. Software libre. <http://upcommons.upc.edu/pfc/bitstream/2099.1/6426/1/SergioLopezMemoria.pdf>.
- Mapa conceptual del software libre. http://upload.wikimedia.org/wikipedia/commons/6/61/Mapa_conceptual_del_software_libre.svg
- Marcos David Kon, El software libre. <http://www.monografias.com/trabajos12/elsoflib/elsoflib.shtml>
- Marcos David Kon, El software libre. <http://www.monografias.com/trabajos12/elsoflib/elsoflib.shtml>
- Maria Lucila, Software libre. <http://www.monografias.com/trabajos71/software-libre/software-libre.shtml>
- Mateu , Carles; Desarrollo de aplicaciones web. Software libre. http://www.sw-computacion.f2s.com/Linux/004-Desarrollo_de_aplicaciones_web.pdf.
- Prieto Daza ,Guillermo. Metodología para el desarrollo de aplicaciones en la plataforma: Linux PDA. Software libre. <http://es.tldp.org/Manuales-LuCAS/doc-tesis-desarrollo-linux-PDAs/thesis.pdf>
- Salcedo Lucio, Francisco David. Líneas de investigación y desarrollo en informática educativa – escuela ciencias informáticas. Software libre. http://www.ueb.edu.ec/uploads/Documentos/Ciencias_educacion/LINEAS_INVESTIGACION_INFORMATICA_EDUCATIVA.pdf.
- SergioN, Software. <http://es.wikipedia.org/wiki/Software>
- Wikipedia, Aplicación Web. http://es.wikipedia.org/wiki/Aplicacion_web
- Wikipedia, Software libre. http://es.wikipedia.org/wiki/Software_libre
- Wikipedia, Software libre. http://es.wikipedia.org/wiki/Software_libre