

*IMPLEMENTACIÓN EN SOFTWARE LIBRE DE LA APLICACIÓN DE  
INVENTARIO, CIRCULACIÓN Y PRÉSTAMO BIBLIOGRÁFICO DEL COLEGIO  
EDUCACIÓN Y VIDA*

*DIANA YAMILE URREGO REYES*

*UNIVERSIDAD DE SAN BUENAVENTURA  
FACULTAD DE INGENIERÍA  
TECNOLOGÍA DE SISTEMAS  
BOGOTÁ  
2007*

*IMPLEMENTACIÓN EN SOFTWARE LIBRE DE LA APLICACIÓN DE  
INVENTARIO, CIRCULACIÓN Y PRÉSTAMO BIBLIOGRÁFICO DEL COLEGIO  
EDUCACIÓN Y VIDA*

*DIANA YAMILE URREGO REYES*

*Monografía para optar al título de Tecnólogo de Sistemas*

*Asesor  
FABIÁN MOJICA  
Docente Académico*

*UNIVERSIDAD DE SAN BUENAVENTURA  
FACULTAD DE INGENIERÍA  
TECNOLOGÍA DE SISTEMAS  
BOGOTÁ  
2007*

*Nota de aceptación*

\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_  
*Presidente del Jurado*

\_\_\_\_\_  
*Jurado*

\_\_\_\_\_  
*Jurado*

*Bogotá D.C., Noviembre 24 de 2007*

*A Dios el realizador de todas mis metas  
A mi familia por su ayuda  
A mi cómplice por su apoyo y colaboración*

*Gracias*

|                                                         |    |
|---------------------------------------------------------|----|
| INTRODUCCIÓN .....                                      | 8  |
| 1. PLANTEAMIENTO DEL PROBLEMA .....                     | 10 |
| 1.1 ANTECEDENTES .....                                  | 10 |
| 1.2 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA .....        | 10 |
| 1.3 JUSTIFICACIÓN .....                                 | 11 |
| 1.4 OBJETIVOS DE LA INVESTIGACIÓN .....                 | 11 |
| 1.4.1 Objetivo General. ....                            | 11 |
| 1.4.2 Objetivos Específicos. ....                       | 11 |
| 1.5 ALCANCES Y LIMITACIONES DEL PROYECTO .....          | 12 |
| 1.5.1 Alcances.....                                     | 12 |
| 1.5.2 Limitaciones. ....                                | 12 |
| 2. MARCO DE REFERENCIA .....                            | 13 |
| 2.1 MARCO TEÓRICO.....                                  | 13 |
| 2.1.1 El Software.....                                  | 13 |
| 2.1.2 Software libre. ....                              | 15 |
| 2.1.2.1 Soluciones de software libre.....               | 18 |
| 2.1.3 Tipos de Licencias .....                          | 19 |
| 2.1.3.1 Clasificación de las licencias de software..... | 21 |
| 2.1.4 Linux.....                                        | 24 |
| 2.1.4.1 Distribuciones de Linux.....                    | 25 |
| 2.1.4.2 Aplicaciones de los sistemas Linux. ....        | 26 |
| 2.1.5 Interfaz de Usuario. ....                         | 27 |
| 2.1.5.1 PHP.....                                        | 28 |
| 2.1.6 Motores de bases de datos .....                   | 32 |
| 2.1.6.1 Nivel Físico .....                              | 32 |
| 2.1.6.2 Nivel Conceptual.....                           | 33 |
| 2.1.7 Software libre en Colombia .....                  | 34 |
| 2.2 MARCO LEGAL O NORMATIVO .....                       | 38 |
| 2.2.1 Libertades del Software Libre.....                | 39 |
| 3. TÉCNICAS DE RECOLECCIÓN DE INFORMACIÓN.....          | 42 |
| 4. DESARROLLO DEL PROYECTO.....                         | 43 |
| 5. CONCLUSIONES.....                                    | 47 |
| 6. RECOMENDACIONES .....                                | 48 |
| BIBLIOGRAFÍA .....                                      | 49 |
| GLOSARIO.....                                           | 50 |
| ANEXOS .....                                            | 53 |

## **LISTA DE TABLAS**

|                                                                                                  |           |
|--------------------------------------------------------------------------------------------------|-----------|
| <i>TABLA 1. COMPARACIÓN DE SISTEMAS ADMINISTRADORES DE BASES DE DATOS<br/>RELACIONALES .....</i> | <i>53</i> |
| <i>TABLA 2. SOPORTE DEL SISTEMA OPERATIVO .....</i>                                              | <i>54</i> |
| <i>TABLA 3. CARACTERÍSTICAS FUNDAMENTALES .....</i>                                              | <i>55</i> |
| <i>TABLA 4. COMPARACIÓN ENTRE SOFTWARE LIBRE Y SOFTWARE PRIVATIVO .....</i>                      | <i>56</i> |

## **LISTA DE ILUSTRACIONES**

|                                                                   |           |
|-------------------------------------------------------------------|-----------|
| <i>ILUSTRACIÓN 1. MAPA CONCEPTUAL DEL SOFTWARE.....</i>           | <i>16</i> |
| <i>ILUSTRACIÓN 2. DIAGRAMA ENTIDAD - RELACIÓN .....</i>           | <i>44</i> |
| <i>ILUSTRACIÓN 3. DISEÑO DE INTERFAZ DE USUARIO .....</i>         | <i>45</i> |
| <i>ILUSTRACIÓN 4. CRONOGRAMA DE DESARROLLO DEL PROYECTO .....</i> | <i>57</i> |



## **INTRODUCCIÓN**

*A la hora de incorporar Tecnologías en la Educación, se presentan diversos inconvenientes, uno de los más importantes es el costo.*

*Los costos asociados a Infraestructura de redes, conectividad, hardware, software, capacitación de personal, instalación y mantenimiento, pueden ser una limitante especialmente para países en desarrollo como Colombia. De los costos mencionados el Hardware y el Software son los más significativos, sin embargo el costo de la infraestructura hardware va en descenso, mientras que algunas empresas desarrolladoras y comercializadoras de software imponen costos a la licencia del mismo cada vez más altos.*

*Esto a su vez ha influido en el aumento excesivo de los índices de piratería de software en el mundo.*

*Sin embargo, existe una alternativa que puede disminuir las barreras de acceso a las nuevas tecnologías a través de una significativa reducción en los costos del software, se trata del software libre.*

*El software libre puede adquirirse sin ningún costo, no hay que pagar por el licenciamiento, este puede ser compartido e instalado en los computadores que sea necesario, generalmente la licencia del software privativo tiene un costo por cada usuario o computador donde sea instalado. Las actualizaciones de software libre se pueden adquirir a un costo despreciable, en contraste, por las actualizaciones del software privativo se debe pagar un precio generalmente alto. Por esta razón, se considera que el software libre es una solución tecnológica económicamente viable.*

*Existe un tipo de software predominante en el mercado, el cual se conoce como software privativo, por el cual, en la mayoría de los casos, se debe pagar un alto costo por su licencia, este generalmente varía dependiendo del uso que se le pretenda dar a dicho software (por ejemplo el costo de la licencia del sistema operativo es diferente si el propósito de implementación es pedagógico o si es administrativo). Por otra parte, este software presenta cierto tipo de restricciones al usuario final del programa ya que no se puede redistribuir copias del software,*

*instalarlo en un número de equipos diferentes al establecido en la licencia, estudiarlo y modificarlo para adaptarlo a las condiciones propias del entorno o de la región, ya que el código fuente del programa no está disponible, lo cual genera altos costos para la institución que lo desea utilizar, como es el caso del colegio Educación y Vida, el cual necesita hacer la actualización de sus equipos, pero el alto costo de las licencias es un gran impedimento para cumplir dicho objetivo.*

## **1. PLANTEAMIENTO DEL PROBLEMA**

### **1.1 ANTECEDENTES**

*La Universidad de San Buenaventura a través de su proyecto integrador como estrategia de proyección social e investigación, desarrolló en el 2006 con los estudiantes de sexto semestre de Ingeniería de Sistemas, la aplicación con la que cuenta en este momento el Colegio Educación y Vida; que se encarga de manejar el movimiento de circulación y préstamo de libros en la biblioteca del colegio.*

*Dicho sistema se encuentra creado en Access 2003 y corre bajo Windows XP, además cuenta con un lector óptico que facilita el préstamo y salida de libros, por medio de un código de barras que llevan en los libros.*

*Debido a los avances tecnológicos el colegio se ve en la necesidad de hacer una actualización del software que utiliza actualmente, esta actualización en cuanto a licenciamiento únicamente podría hacerse a Windows Vista, lo que significa un alto costo, no solo en licenciamiento, si no, en actualización de equipos debido a los requerimientos necesarios para dicho sistema operativo.*

*Los equipos del colegio deben ser actualizados y por ende el software para el manejo de biblioteca debe ser renovado, pero el colegio no cuenta con los fondos suficientes para llevar a cabo esta actualización.*

### **1.2 DESCRIPCIÓN Y FORMULACIÓN DEL PROBLEMA**

*El Colegio Educación y Vida necesita hacer la actualización de equipos y por ende del Sistema Operativo con que cuenta actualmente, pero no puede pagar los altos costos de licenciamiento del software privativo.*

*Por tal motivo se hace necesaria una aplicación para el manejo de la Biblioteca, que no solamente evite los altos costos de implementación, si no, además permita la utilización de los recursos actuales.*

*Esto no solo permitirá a la institución un ahorro considerable de dinero, si no, que a su vez, una mejora en la implementación de tecnologías que permitan el aprovechamiento de los recursos existentes por parte de los estudiantes, no obstante podemos confirmar la siguiente pregunta:*

*¿Cómo implementar técnica y funcionalmente un aplicativo de inventario, circulación y préstamo de libros de la biblioteca, sin que esto represente un alto costo económico para la Institución?*

### **1.3 JUSTIFICACIÓN**

*Debido a la falta de recursos del colegio para hacer la actualización de sistema operativo existente, se propone como solución el cambio a software libre, lo cual no representara altos costos para el colegio y aporta a la implementación de nuevas tecnologías informáticas, y esto permitirá no solo un avance tecnológico para la institución, si no , además es un aporte importante para lograr la certificación ISO 9001, la cual es uno de los objetivos propuestos por el colegio para el año en curso, lo que favorecerá a un aumento en los índices de calidad en la educación para dicho establecimiento, además de ser un compromiso adquirido por la Universidad de San Buenaventura, en su plan de proyección social.*

### **1.4 OBJETIVOS DE LA INVESTIGACIÓN**

#### **1.4.1 Objetivo General.**

*Implementar un aplicativo para el control de inventario, circulación y préstamo de documentos bibliográficos, para la biblioteca del colegio Educación y Vida utilizando software libre.*

#### **1.4.2 Objetivos Específicos.**

- 1. Analizar el aplicativo de inventario, circulación y préstamo existente.*
- 2. Diseñar el modelado de la base de datos a utilizar*
- 3. Desarrollar la interfaces graficas del usuario.*
- 4. Desarrollar el software libre a utilizar.*

## **1.5 ALCANCES Y LIMITACIONES DEL PROYECTO**

### **1.5.1 Alcances.**

*Este proyecto exalta la importancia de la tecnología y un adelanto en el desarrollo para ponerlo a disposición de administrativos, docentes, estudiantes y demás agentes que participan en el servicio de la biblioteca del Colegio Educación y Vida. Por lo tanto, es importante garantizar que la calidad y seguridad del servicio de la biblioteca, avalada con las nuevas tecnologías; y teniendo claro que se modificaran los escenarios y las percepciones actuales, una vez que los actores se adapten, los resultados pueden igualarse o superarse en comparación con la aplicación actual.*

*El proyecto culmina con el desarrollo del aplicativo de inventario, circulación y préstamo de libros corriendo bajo software libre; El producto final del proyecto se entregara a la Universidad de San Buenaventura y no al Colegio de Educación y Vida.*

### **1.5.2 Limitaciones.**

*La limitación principal para la culminación e implantación de este proyecto es la entrega a tiempo de la información requerida para el cargue de las bases de datos del aplicativo, además, la terminación de la red implantada por los estudiantes de Ingeniería de Sistemas, para la realización de las pruebas de funcionalidad.*

## **2. MARCO DE REFERENCIA**

### **2.1 MARCO TEÓRICO**

*Para la realización de este proyecto se tuvieron en cuenta diferentes temas que tienen prelación a la hora de crear una aplicación y sobre todo una aplicación basada en software libre, teoría que se puede encontrar en los apartes que se encuentran a continuación:*

#### **2.1.1 El Software.**

*Para la realización de cualquier desarrollo es de gran importancia el conocimiento de algo tan básico como el software lo cual podemos definir como una producción inmaterial del cerebro humano y tal vez una de las estructuras más complicadas que la humanidad conoce. De hecho, los expertos en computación aún no entienden del todo cómo funciona, su comportamiento, sus paradojas y sus límites. Básicamente, el software es un plan de funcionamiento para un tipo especial de máquina, una máquina "virtual" o "abstracta". Una vez escrito mediante algún lenguaje de programación, el software se hace funcionar en ordenadores, que temporalmente se convierten en esa máquina para la que el programa sirve de plan. El software permite poner en relación al ser humano y a la máquina y también a las máquinas entre sí. Sin ese conjunto de instrucciones programadas, los ordenadores serían objetos inertes, como cajas de zapatos, sin capacidad siquiera para mostrar algo en la pantalla.*

*Los ordenadores sólo procesan lenguaje binario, pero para las personas este no es un modo válido de comunicarse. Si bien en los tiempos heroicos de los primeros ordenadores no les quedaba otro remedio que hacerlo, los programadores hace mucho que no escriben su código en lenguaje binario (denominado técnicamente "código-máquina"), pues es terriblemente tedioso, improductivo y muy sujeto a errores. Hace tiempo que los programadores escriben las instrucciones que ha de ejecutar el procesador de la máquina mediante lenguajes formales, llamados "de alto nivel", bastante cercanos al inglés, si bien con rígidas reglas sintácticas que lo asemejan a los lenguajes lógico-formales. Esto facilita enormemente la tarea de escribir programas pero, para que esas instrucciones sean comprensibles para el procesador, deben ser convertidas antes a código-máquina. Esa conversión se realiza cómodamente con programas*

*especiales, llamados compiladores. A lo que escribe el programador se le denomina ``código-fuente". Al resultado de la ``conversión" (compilación) en lenguaje-máquina, se le denomina ``código-objeto", ``binarios" o ``ficheros ejecutables". En principio, al usuario común sólo le importa este último nivel, los ``binarios", pero conviene tener clara la distinción entre fuentes y binarios pues es clave para entender el empeño de los partidarios del software libre en disponer de las fuentes.*

*El termino free, traducido al castellano, significa tanto libre como gratis, por eso muchas veces suelen confundirse el freeware con el software libre aunque entre ambos existen notables diferencias.<sup>1</sup>*

---

<sup>1</sup> <http://biblioweb.sindominio.net/telematica/softlibre/node1.html>

### **2.1.2 Software libre.**

*¿Quién no ha oído hablar del software libre? El más conocido y con mayor publicidad es el sistema operativo Linux. Pero poco se habla sobre que es realmente el software libre y menos sobre el beneficio que tiene este tipo de software en el mundo empresarial. (Ver tabla 4)*

*El termino “software libre” es la traducción del ingles “free software”; en ingles el termino “free” conlleva confusión, ya que significa tanto “libre” como “gratis”. Por ello fue necesario aclarar que la palabra “libre” (en ingles la palabra free) no se refiere a términos de costo económico, sino al termino libertad (del ingles freedom). Con este significado se ha utilizado el termino desde los años 80, apareciendo la primera definición documentada en el año 1989. Existen cuatro “libertades” que definen el software libre:*

*La libertad de usar el programa, con cualquier propósito.*

*La libertad de estudiar cómo funciona el programa, y adaptarlo a tus necesidades.*

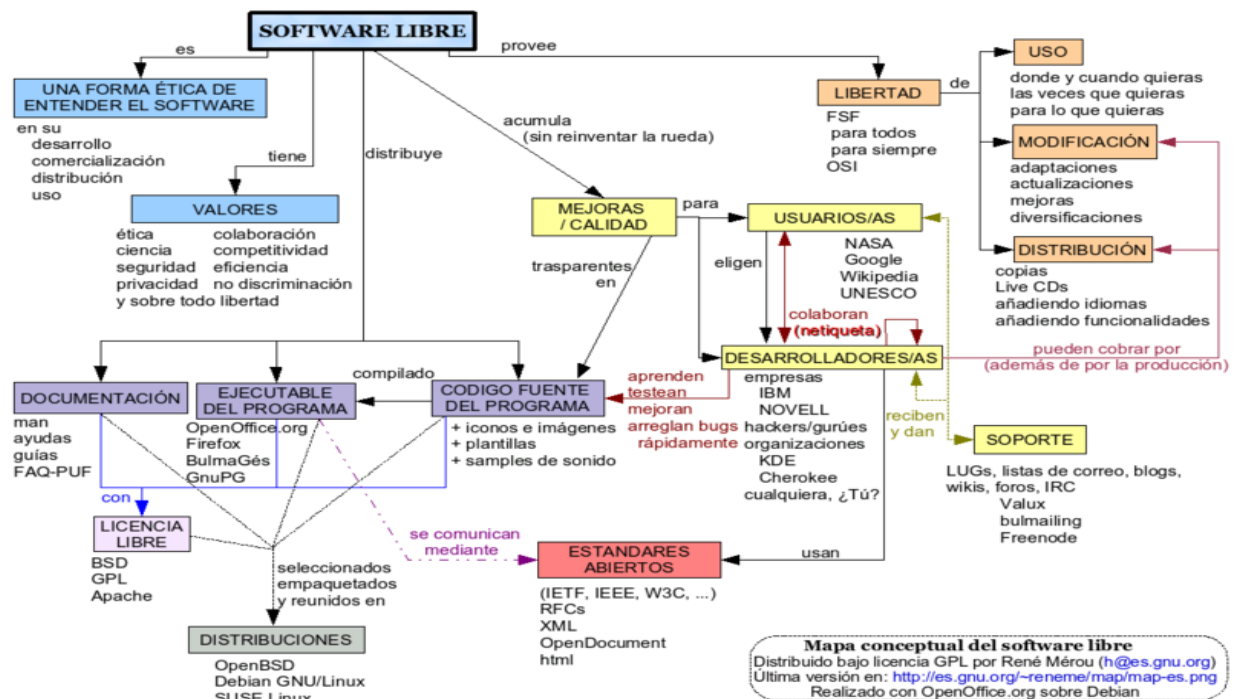
*La libertad de distribuir copias, con lo que puedes ayudar a tu vecino.*

*La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.*

*Estas libertades son derechos, no obligaciones, aunque si se respetan estas libertades en beneficio de la sociedad puede que en algún momento obliguen al individuo. Toda persona es libre de no usar estas libertades, pero también podría elegir el uso de todas ellas.*

*Es importante que el software libre no excluya el uso comercial ni la distribución comercial, ya que no es software gratuito. Por ejemplo un número creciente de empresas basa su negocio completamente o al menos parcialmente en el software libre. El software libre hace legal proveer ayuda y asistencia, pero no lo hace imprescindible.*

En este sentido el software libre, es a menudo, la mejor opción para la empresa. Ya no en términos económicos, debido a que por ejemplo la implementación, ayuda o asistencia para software libre generalmente es más barata, sino en términos de control y/o despersonalización. Pongamos el ejemplo de una empresa que necesita una herramienta de formación a distancia para sus trabajadores, la implementación, desarrollo y soporte de dicha herramienta sería muy costoso si esta es desarrollada de cero, en cambio el costo de un programa libre sería mínimo, debido a que este ya sería un programa final fiable que simplemente se tenga que adaptar a las necesidades particulares de la empresa.



**Ilustración 1. Mapa conceptual del software**

Actualmente el gran problema de la implementación de software libre en las empresas es el miedo a que estos programas dejen de mejorarse o desarrollarse. Pero esto es una falsa creencia, ya que la filosofía del software libre impide que nunca se de este punto, en cambio esta situación crítica puede darse con el software no libre. Se puede tomar como ejemplo un programa de atención a clientes online, puede ocurrir que el equipo de creadores y desarrolladores del programa finalicen el desarrollo y mejora de este; es muy posible que ante esta situación otros desarrolladores continúen con el desarrollo y mejora del programa. En cambio si se diera la misma situación de punto final con software no libre no se

*podría continuar el desarrollo y mejora por parte de terceras personas debido a que el código del programa no sería abierto o público.*

*Pero el software libre es mucho más que el derecho de los programadores y de los hackers a disponer de las fuentes del código: significa también la libertad de copiar y redistribuir esos programas. Esos derechos, o su ausencia, condicionan a cualquiera que use un ordenador y han configurado la industria del software y de la informática tal y como la conocemos hoy día.*

### **2.1.2.1 Soluciones de software libre.**

*El software libre es un fenómeno que está llamado a revolucionar modelos de negocio de la industria del software. No es este el momento de analizar con detalle el mundo del software libre, ni las plataformas tecnológicas, ni la tipología de licencias existente, ya que para todo ello son muy abundantes los recursos de información disponibles. Para los objetivos de este texto es suficiente con establecer los criterios básicos que deben cumplir las herramientas reseñadas: Ofrecer el código fuente de la aplicación.*

*Distribuirse bajo alguna de las licencias consideradas de referencia (véanse las licencias disponibles en Open Source Initiative .*

*Poder ser modificadas, copiadas y distribuidas libremente, respetando los términos establecidos en la licencia respectiva.*

### **2.1.3 Tipos de Licencias .**

*Una licencia es aquella autorización formal con carácter contractual que un autor de un software da a un interesado para ejercer "actos de explotación legales". Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciatario. Desde el punto de vista del software libre, existen distintas variantes del concepto o grupos de licencias.<sup>2</sup>*

*Las libertades definidas anteriormente están protegidas por licencias de software libre, de las cuales una de las más utilizadas es la Licencia Pública General GNU (GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más*

---

<sup>2</sup> ProArgentina(Editor). Software/ América Latina.  
Argentina: El Cid Editor, 2005. p 3.

*restrictivos de la propia GNU GPL. Esto hace que no sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL.*

*Licencias estilo BSD, llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles.*

*Puede argumentarse que esta licencia asegura “verdadero” software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que puede decidir incluso redistribuirlo como no libre. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre.*

*Licencias estilo MPL y derivadas, Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL es Software Libre y promueve eficazmente la colaboración evitando el efecto “viral” de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU. En el espectro de las licencias de software libre se la puede considerar adyacente a la licencia estilo BSD, pero perfeccionada.*

*Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica no restringe ninguno de los derechos otorgados a*

*los usuarios de la versión copyleft. También podría retirar todas las licencias de software libre anteriormente otorgadas, pero esto obligaría a una indemnización a los titulares de las licencias en uso. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias "copyleft", no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no ha derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.*

### **2.1.3.1 Clasificación de las licencias de software.**

*A continuación se encuentran los diferentes tipos de licencia existentes para el uso del software libre, cabe anotar que este proyecto no tendrá ningún tipo de licenciamiento.*

*Según los derechos que cada autor se reserva sobre su obra*

#### **Licencia de software libre sin protección heredada**

*Se puede crear una obra derivada sin que ésta tenga obligación de protección alguna. Muchas licencias pertenecen a esta clase, entre otras:*

- *Academic Free License v.1.2.*
- *Apache Software License v.1.1.*
- *Artistic.*
- *Attribution Assurance license.*
- *BSD License.*
- *MIT License.*
- *University of Illinois/NCSA Open Source License;*
- *W3C Software Notice and License.*
- *Calistornetion Advance License.*

#### **Licencia de software libre con protección heredada**

*Algunas restricciones se aplican a las obras derivadas. Entre las licencias de esta categoría están:*

- *Artistic License.*
- *Common Public License v.1.0.*

- GNU General Public License v.2.0.
- GNU Lesser General Public License v.2.1.
- Mozilla Public License

***Licencia de software semilibre***

*Se permite uso, copia, modificación o redistribución permitida sin fines de lucro.*

***Licencia de software no libre***

*Se protege contra uso, copia o redistribución.*

***Licencia de software semilibre antagónico***

*Este se puede utilizar por varios usuarios con el uso de una sola licencia y usuario.*

*Según su destinatario*

***Licencia de Usuario Final***

*license agreement, es una licencia en que se permite sólo el uso del mismo.*

*En este tipo de contrato el dueño de los derechos de un producto insta al usuario final de este a que reconozca tener conocimiento de las restricciones de derechos de autor (copyright), patentes, etc. y acepte un contrato general de uso del mismo para poder hacer uso del mismo.*

*Las compañías utilizan este tipo de contratos para tener un mayor control del uso de sus productos y como un documento único de aceptación cuando los componentes del producto estén regulados por más de un contrato o licencia.*

*A pesar de que este tipo de contratos se encuentran regulados en la mayor parte de las veces al menos por la legislación del país de origen del producto y a veces también por las legislaciones locales, este tipo de contratos han sido muy atacados por diversas razones entre las que destacan:*

*El conocimiento del contenido de los contratos es difícil antes de la compra del producto ya que las cajas de los productos raramente contienen una copia completa del mismo, dándose que el comprador en la mayor parte de las ocasiones conoce su contenido después de la compra.*

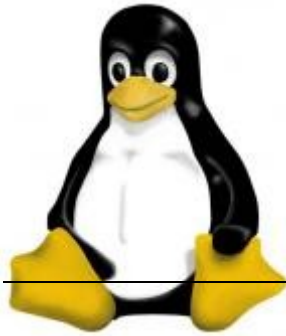
*En ocasiones se exige al usuario renunciar a realizar reclamos o demandas legales por diversos motivos tales como posibles daños producidos por el producto (o fallos en el) o aceptar la restitución de estos en la forma y monto que la compañía decida.*

*Este tipo de acuerdo expresa que tipos de usos se pueden dar y cuales no al producto ya que quien lo compra no es legalmente en ninguna forma dueño del producto sino de una licencia para su uso., considerándose esto último por algunas personas una limitación a los derechos del consumidor.*

*Este tipo de acuerdos son unilaterales pues el usuario no tiene más opción que aceptar o rechazar el contenido del mismo (en algunos países existen organizaciones de protección al consumidor que previamente los autorizan).*

#### **2.1.4 Linux.**

*Es la denominación de un sistema operativo tipo-Unix y el nombre de un núcleo. Es uno de los paradigmas más prominentes del software libre y del desarrollo del código abierto<sup>3</sup>, cuyo código fuente está disponible públicamente y cualquier persona puede libremente usarlo, estudiarlo, redistribuirlo y, con los conocimientos informáticos adecuados, modificarlo. (Ver Tabla 2)*



*Los primeros sistemas Linux se originaron en 1992, al combinar utilidades de sistema y librerías del proyecto GNU con el núcleo Linux, completando un sistema también conocido como GNU/Linux. Desde fines de 1990 Linux ha obtenido el apoyo de diversas empresas multinacionales del*

<sup>3</sup> [http://es.wikipedia.org/wiki/Linux#\\_note-5#\\_note-5](http://es.wikipedia.org/wiki/Linux#_note-5#_note-5)

*mundo de la informática, tales como IBM, Sun Microsystems, Hewlett-Packard y Novell.*

*Linux es usado como sistema operativo en una amplia variedad de plataformas de hardware y computadores, incluyendo los computadores de escritorio (PCs x86 y x86-64, y Macintosh y PowerPC), servidores, supercomputadores, mainframes, y dispositivos empotrados así como teléfonos celulares.*

*La marca Linux (Número de serie: 1916230) pertenece a Linus Torvalds y se define como "un sistema operativo para computadoras que facilita su uso y operación".*

*Linux se refiere estrictamente al núcleo Linux, pero es comúnmente utilizado para describir al sistema operativo tipo Unix (que implementa el estándar POSIX), que utiliza primordialmente filosofía y metodologías libres (también conocido como GNU/Linux) y que está formado mediante la combinación del núcleo Linux con las bibliotecas y herramientas del proyecto GNU y de muchos otros proyectos/grupos de software (libre o no libre).*

*La expresión "Linux" es utilizada para referirse a las distribuciones GNU/Linux, colecciones de software que suelen contener grandes cantidades de paquetes además del núcleo. El software que suelen incluir consta de una enorme variedad de aplicaciones, como: entornos gráficos, suites ofimáticas, servidores web, servidores de correo, servidores FTP, etcétera. Coloquialmente se aplica el término "Linux" a éstas. Algunas personas opinan que es incorrecto denominarlas distribuciones Linux, y proponen llamarlas sistema GNU/Linux. Otras personas opinan que los programas incluidos proceden de fuentes tan variadas que proponen simplificarlo denominándolo simplemente a "Linux".*

#### **2.1.4.1 Distribuciones de Linux.**

*Una distribución es un conjunto de aplicaciones reunidas por un grupo, empresa o persona para permitir instalar fácilmente un sistema Linux. Es un sabor de Linux. En general se destacan por las herramientas para configuración y sistemas de paquetes de software a instalar.*

*Existen numerosas distribuciones Linux (también conocidas como "distros"), ensambladas por individuos, empresas y otros organismos. Cada distribución*

*puede incluir cualquier número de software adicional, incluyendo software que facilite la instalación del sistema. La base del software incluido con cada distribución incluye el núcleo Linux y las herramientas GNU, al que suelen adicionarse también varios paquetes de software.*

*Las herramientas que suelen incluirse en la distribución de este sistema operativo se obtienen de diversas fuentes, incluyendo de manera importante proyectos de código abierto o libre, como el GNU y el BSD o el KDE. Debido a que las herramientas de software libre que en primera instancia volvieron funcional al núcleo de Linux provienen del proyecto GNU que desde 1983 había liberado software que pudo ser usado en el proyecto de Linux de 1991, Richard Stallman (fundador del proyecto GNU) pide a los usuarios que se refieran a dicho sistema como GNU/Linux. A pesar de esto, la mayoría de los usuarios continúan llamando al sistema simplemente "Linux" y las razones expuestas por Richard Stallman son eterno motivo de controversia. La mayoría de los sistemas "Linux" incluyen también herramientas procedentes de BSD y de muchos otros proyectos como Mozilla, Perl, Ruby, Python, PostgreSQL, MySQL, Xorg, casi todas con licencia GPL o compatibles con ésta (LGPL, MPL) otro aporte fundamental del proyecto GNU.*

*Usualmente se utiliza la plataforma XFree86 o la X.Org para sostener interfaces gráficas.*

#### **2.1.4.2 Aplicaciones de los sistemas Linux.**

*Con la adopción por numerosas empresas fabricantes de PCs, muchas computadoras son vendidas con distribuciones GNU/Linux pre-instaladas, y "GNU/Linux" ha comenzado a tomar su lugar en el vasto mercado de las computadoras de escritorio.*

*Con entornos de escritorio, "GNU/Linux" ofrece una interfaz gráfica alternativa a la tradicional interfaz de línea de comandos de Unix. Existen en la actualidad numerosas aplicaciones gráficas, ya sean libres o no, que ofrecen funcionalidad que está permitiendo que GNU/Linux se adapte como herramienta de escritorio.*

*Algunas distribuciones permiten el arranque de Linux directamente desde un disco compacto (llamados LiveCDs) sin modificar en absoluto el disco duro de la computadora en la que se ejecuta Linux. Para este tipo de distribuciones, en general, los archivos de imagen (archivos ISO) están disponibles en Internet para su descarga.*

*Otras posibilidades incluyen iniciar el arranque desde una red (ideal para sistemas con requerimientos mínimos) o desde un disco flexible o disquete o de unidades de almacenamiento USB.*

### **2.1.5 Interfaz de Usuario.**

*La interfaz de usuario es la forma en que los usuarios pueden comunicarse con una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Sus principales funciones son:*

*Manipulación de archivos y directorios  
Herramientas de desarrollo de aplicaciones  
Comunicación con otros sistemas  
Información de estado  
Configuración de la propia interfaz y entorno  
Intercambio de datos entre aplicaciones  
Control de acceso  
Sistema de ayuda interactivo.*

*Nos encontramos con dos tipos de interfaz de usuario:*

*Interfaces alfanuméricas (intérpretes de mandatos).*

*Interfaces gráficas de usuario (GUI, Graphics User Interfaces), las que permiten comunicarse con el ordenador de una forma muy rápida e intuitiva.*

*Y pueden ser de hardware o de software:*

*En el primer caso se trata de un conjunto de dispositivos que permiten la interacción hombre-máquina, de modo que permiten ingresar y tomar datos del ordenador.*

*También están las interfaces de software que son programas o parte de ellos que permiten expresar nuestros deseos al ordenador.*

*Sin embargo, no siempre se trata de interfaces intuitivas tal como el caso de las interfaces de línea de órdenes (CLI), que se encuentran por ejemplo en algunos sistemas operativos como los NOS de los Routers o algunos shell de Unix, DOS, etc.*

#### **2.1.5.1 PHP.**

*PHP es un lenguaje interpretado del lado del servidor que surge dentro de la corriente denominada código abierto (open source). Se caracteriza por su potencia, versatilidad, robustez y modularidad. Al igual que ocurre con tecnologías similares, los programas son integrados directamente dentro del código HTML.*

*Por otro lado, los programas en ASP resultan más lentos y pesados, y también menos estables. En los entornos Microsoft la ventaja de ASP es que los servidores web de Microsoft soportan directamente ASP sin necesidad de ninguna instalación adicional. Señalar también la existencia de herramientas que permiten convertir programas desarrollados en ASP al lenguaje PHP, una de las más conocidas es asp2php.*

*Comparando el lenguaje PHP con el lenguaje Perl, utilizado habitualmente en la programación CGI, puede decirse que PHP fue diseñado para desarrollo de scripts*

*orientados a web, mientras que Perl fue diseñado para hacer muchas más cosas y debido a esto, se hace muy complicado. La sintaxis de PHP es menos confusa y más estricta, pero sin perder la flexibilidad.*

*En comparación con ColdFusion, PHP es más rápido y eficiente para tareas complejas de programación, además PHP resulta más estable y usa una menor cantidad de recursos. Por el contrario, ColdFusion posee un mejor gestor de errores, un buen motor de búsquedas, abstracciones de bases de datos y un gran número de funcionalidades para el procesamiento de fechas. Finalmente, ColdFusion no está disponible para todas las plataformas.*

*En definitiva, PHP es uno de los lenguajes más utilizados actualmente en el desarrollo de aplicaciones web y viene experimentando un constante crecimiento en su nivel de utilización en Internet. Este libro trata de humildemente contribuir a continuar con el proceso de difusión de esta tecnología.<sup>4</sup>*

---

<sup>4</sup> Cobo, Ángel. PHP y MySQL: tecnologías para el desarrollo de aplicaciones web. España: Ediciones Díaz de Santos, 2007. p 23.

## *Versiones*

### *PHP 3.2.4.3*

*Dos programadores israelíes del Technion, Zeev Suraski y Andi Gutmans, reescribieron el analizador sintáctico (parser en inglés) en el año 1997 y crearon la base del PHP3, cambiando el nombre del lenguaje a la forma actual. Inmediatamente comenzaron experimentaciones públicas de PHP3 y fue publicado oficialmente en junio del 1998.*

*Para 1999, Suraski y Gutmans reescribieron el código de PHP, produciendo lo que hoy se conoce como Zend Engine o motor Zend, un portmanteau de los nombres de ambos, Zeev y Andi. También fundaron Zend Technologies en Ramat Gan, Israel.*

---

#### *PHP 4*

*En mayo de 2000 PHP 4 fue lanzado bajo el poder del motor Zend Engine 1.0. La última versión de PHP 4 disponible en febrero de 2007 es la 4.4.5. El soporte a PHP 4 continúa activo lanzando parches de seguridad para aquellas aplicaciones que lo requieren.*

#### *PHP 5*

*El 13 de julio de 2004, fue lanzado PHP 5, utilizando el motor Zend Engine II (o Zend Engine 2). La versión más reciente de PHP es la 5.2.1, que incluye todas las ventajas que provee el nuevo Zend Engine 2 como:*

*Soporte sólido para Programación Orientada a Objetos ( o OOP) con PHP Data Objects*

*Mejoras de rendimiento*

*Mejor soporte para MySQL con extensión completamente reescrita*

*Mejor soporte a XML ( XPath, DOM... )*

*Soporte nativo para SQLite*

*Soporte integrado para SOAP*

*Iteradores de datos*

*Excepciones de errores*

*La última versión a Mayo 2007 es la 5.2.2.*

#### *PHP 6*

*Está previsto el lanzamiento en breve de la rama 6 de PHP, cuando se lance esta nueva versión, quedarán tres ramas activas en desarrollo ( PHP 4, 5 y 6).*

*Las diferencias que encontraremos frente a PHP 5 son:*

*Soportará Unicode*

*Limpieza de funcionalidades obsoletas como register\_globals, safe\_mode...*

*PECL*

*Mejoras en orientación a objetos*

### **2.1.6 Motores de bases de datos.**

*Las Base de Datos son un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en máquina accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo. Surgen desde mediados de los años sesenta la historia de las bases de datos, en 1970 Codd propuso el modelo relacional, este modelo es el que ha marcado la línea de investigación por muchos años, ahora se encuentran los modelos orientados a objetos. (Ver Tabla 3)*

*Independencia de datos y tratamiento.*

*Cambio en datos no implica cambio en programas y viceversa (Menor coste de mantenimiento).*

*Coherencia de resultados.*

*Reduce redundancia :*

*Acciones lógicamente únicas.*

*Se evita inconsistencia.*

*Mejora en la disponibilidad de datos*

*No hay dueño de datos (No igual a ser públicos).*

*Ni aplicaciones ni usuarios.*

*Guardamos descripción (Idea de catálogos).*

*Cumplimiento de ciertas normas.*

*Restricciones de seguridad.*

*Accesos (Usuarios a datos).*

*Operaciones (Operaciones sobre datos).*

*Otras ventajas:*

*Más eficiente gestión de almacenamiento.*

*Los SBD pueden ser estudiados desde 3 niveles distintos:*

#### **2.1.6.1 Nivel Físico.**

*Es el nivel real de los datos almacenados. Es decir como se almacenan los datos, ya sea en registros, o como sea. Este nivel es usado por muy pocas personas que deben estar cualificadas para ello. Este nivel lleva asociada una representación de los datos, que es lo que denominamos Esquema Físico.*

### **2.1.6.2 Nivel Conceptual**

*Es el correspondiente a una visión de la base de datos desde el punto de visto del mundo real. Es decir tratamos con la entidad u objeto representado, sin importarnos como está representado o almacenado. Este nivel lleva asociado el Esquema Conceptual.*

*Los 3 niveles anteriores, componen lo que conocemos como arquitectura de base de datos a 3 niveles.*

*A menudo el nivel físico no es facilitado por muchos DBMS,<sup>5</sup> esto es, no permiten al usuario elegir como se almacenan sus datos y vienen con una forma estándar de almacenamiento y manipulación de los datos.*

---

<sup>5</sup> DBMS - Sistema de gestión de bases de datos

### **2.1.7 Software libre en Colombia.**

*En Colombia el sector de la Tecnología de Información es de alto crecimiento y dinamismo, teniendo actualmente una tasa cercana al 8%, mostrando esto una oportunidad importante.<sup>6</sup>*

*Se estima que para el 2008 el software en Colombia representará un negocio de u\$ s 345 millones.<sup>7</sup>*

*El software libre juega un factor importante en gobierno, ya que este país adhirió a la Comunidad de Software Libre.*

---

<sup>6</sup> Gil Rivera, Maria del Carmen. La base de datos: importancia y aplicación en educación. México: Red Perfiles Educativos, 2006. p 7.

<sup>7</sup> Gil Rivera, Maria del Carmen. La base de datos: importancia y aplicación en educación. México: Red Perfiles Educativos, 2006. p 7.

*La producción Colombiana es de buena calidad con capacidades de abastecer la demanda. En el caso del software importado tienen generalmente una barrera alta de localización, y es fundamental la asociación local. Dentro del plano de las importaciones de software, la competencia proviene principalmente de Estados Unidos con un 70%, seguido por Alemania, Reino Unido, México y Canadá.*

*Las áreas de demanda pueden establecerse en el sector bancario, en sistemas de gestión bancaria, transacciones interbancarias, salud, en historia clínicas, gestión hospitalaria, administración sanitaria y educación, en todos los niveles.<sup>8</sup>*

*Otras áreas de alto dinamismo económico que consumen productos de software son las telecomunicaciones, energía y agronegocios.<sup>9</sup>*

---

<sup>8</sup> Gil Rivera, Maria del Carmen. La base de datos: importancia y aplicación en educación. México: Red Perfiles Educativos, 2006. p 7.

<sup>9</sup> Fedesoft

*Actualmente Colombia cuenta con un núcleo empresarial conformado por más de 4000 empresas orientadas al sector de TI; se trata de un sector importante de la economía actual colombiana, el cual emplea a más de 75 mil personas y esta conformado mayormente por el sector PyME. Mientras que el sector específicamente de software factura alrededor de 190 millones de dólares y las empresas estrictamente desarrolladoras de software conforman un número muy inferior dentro del sector de TI, alcanzando a las 800, con una planta de empleo de 13 mil profesionales.<sup>10</sup>*

*La cámara que agrupa el sector de TI en Colombia denominada FEDESOFIT agrupa a más de 165 miembros activos.*

*En Colombia no se logró aún desarrollar una industria de tecnología de información y software con una participación importante de las exportaciones, las que en el año 2001 registraron alrededor de 22 millones de dólares, con destino principalmente a sus países limítrofes Ecuador y Venezuela.*

---

<sup>10</sup> Fuente IDC

*El objetivo de enfocar a los mercados de países limítrofes se deslinda del Plan Exportador 1999-2009 elaborado por el Ministerio de Comercio Exterior de este país.<sup>11</sup>*

---

<sup>11</sup> ProArgentina(Editor). Software/ América Latina.  
Argentina: El Cid Editor, 2005. p 16.

## **2.2 MARCO LEGAL O NORMATIVO**

*En cuanto al marco normativo, cabe anotar que este trabajo será realizado bajo software libre, por lo tanto no existe ningún tipo de normatividad, ya que este software se trabaja sin ningún tipo de restricción o licenciamiento; por tal motivo se puede exponer en cambio la normatividad de propiedad intelectual, la cual se define como el derecho de apropiación que se ejecuta sobre las producciones del intelecto humano.*

*Al hablar de la propiedad intelectual del software se debe hablar de derechos de autor, los cuales amparan toda obra o desarrollo de esta categoría, estos derechos los obtiene el autor o creador de la aplicación en el momento en que lo registra ante la entidad de derechos de autor y permiten no solo el reconocimiento de la obra como propia, si no, además, una protección efectiva ante la ley en caso de reproducción del material.*

*Este registro se efectúa ante la Dirección Nacional de Derechos de Autor, la cual es una unidad administrativa del Ministerio del Interior, con una copia del software y el diligenciamiento de un formulario de solicitud para el tipo de obra que para el caso sería software.*

*Una obra registrada con estos derechos queda protegida en todos los países del convenio de Berna y tiene una duración de 50 años.*

*La normatividad que rige la propiedad intelectual es la siguiente:*

- *Ley No. 23 del 28 de Enero de 1982 sobre Derechos de Autor. Dirección Nacional del Derecho de Autor. Ministerio del Interior.*
- *Decisión 344 del 21 de Octubre de 1993, por la cual se adopta el Régimen Común sobre Propiedad Industrial en los países miembros del Acuerdo de Cartagena. Comisión del Acuerdo de Cartagena.*
- *Decisión 345 del 29 de Octubre de 1993, por la cual se adopta el Régimen Común de Protección a los Derechos de Autor de los Obtentores de Variedades Vegetales. Comisión del Acuerdo de Cartagena.*
- *Decisión 351 del 17 de Diciembre de 1993, por la cual se adopta el Régimen Común de Derechos de Autor y Derechos Conexos. Comisión del Acuerdo de Cartagena.*

- *Decreto No. 117 del 14 de Enero de 1994, por el cual se reglamenta la Decisión 344 de la Comisión del Acuerdo de Cartagena. Ministerio de Desarrollo Económico*
- *Decreto 533 del 8 de Marzo de 1994, por el cual se reglamenta el Régimen Común de Protección de Derechos de los Obtentores de Variedades Vegetales. Ministerio de Agricultura.*

### **2.2.1 Libertades del Software Libre**

*De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades:*

*"libertad 0", ejecutar el programa con cualquier propósito (privado, educativo, público, comercial, militar, etc.)*

*"libertad 1", estudiar y modificar el programa (para lo cual es necesario poder acceder al código fuente)*

*"libertad 2", copiar el programa de manera que se pueda ayudar al vecino o a cualquiera*

*"libertad 3", Mejorar el programa y publicar las mejoras*

*Es importante señalar que las libertades 1 y 3 obligan a que se tenga acceso al código fuente. La "libertad 2" hace referencia a la libertad de modificar y redistribuir el software libremente licenciado bajo algún tipo de licencia de software libre que beneficie a la comunidad.*

*Ciertos teóricos usan este cuarto punto (libertad 3) para justificar parcialmente las limitaciones impuestas por la licencia GNU GPL frente a otras licencias de software libre, sin embargo el sentido original es más libre, abierto y menos restrictivo que el que le otorga la propia GNU GPL.*

*La licencia GNU GPL posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Y añade que si se reutiliza en un mismo programa código "A" licenciado bajo licencia GNU GPL y código "B" licenciado bajo otro tipo de licencia libre, el código final "C", independientemente de la cantidad y calidad de cada uno de los códigos "A" y "B", debe estar bajo la licencia GNU GPL.*

*En la práctica esto hace que las licencias de software libre se dividan en dos grandes grupos, aquellas que pueden ser mezcladas con código licenciado bajo GNU GPL (y que inevitablemente desaparecerán en el proceso, al ser el código resultante licenciado bajo GNU GPL) y las que no lo permiten al incluir mayores u otros requisitos que no contemplan ni admiten la GNU GPL y que por lo tanto no pueden ser enlazadas ni mezcladas con código gobernado por la licencia GNU GPL.*

*Esta situación de incompatibilidad, que podría ser resuelta en la próxima versión 3.0 de la licencia GNU GPL (en desarrollo), causa en estos momentos graves perjuicios a la comunidad de programadores de software libre, que muchas veces no pueden reutilizar o mezclar códigos de dos licencias distintas, pese a que las libertades teóricamente lo deberían permitir.*

*El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza estas cuatro libertades. Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación y redistribución para el dueño del copyright; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados.*

*La definición de software libre no contempla el asunto del precio; un eslogan frecuentemente usado es "libre como en libertad, no como en cerveza gratis" o en inglés "Free as in freedom, not as in free beer" (aludiendo a la ambigüedad del término inglés "free"), y es habitual ver a la venta CDs de software libre como distribuciones Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de software libre —por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio, etc—.*

*Para evitar la confusión, algunas personas utilizan los términos "libre" (Libre software) y "gratis" (Gratis software) para evitar la ambigüedad de la palabra inglesa "free". Sin embargo, estos términos alternativos son usados únicamente dentro del movimiento del software libre, aunque están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término open source software (software de código abierto, también llamado de fuentes abiertas). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente en los aspectos técnicos.*

*En un intento por aunar los mencionados términos que se refieren a conceptos semejantes, se está extendiendo el uso de la palabra "FLOSS" con el significado de "Free - Libre - Open Source Software" e, indirectamente, también a la comunidad que lo produce y apoya.*

### **3. METODOLOGÍA**

#### **3.1. ENFOQUE DE LA INVESTIGACIÓN**

*Este proyecto constituye una investigación de tipo empírico – analítico ya que, analizando el proceso de inventario, circulación y préstamo, se ha determinado que existen varios puntos de mejora como lo son el control, la seguridad y la sistematización del mismo. Estas mejoras se lograrán mediante la realización de una interfaz que permita la solicitud de libros, revistas; entre otros, en la biblioteca del Colegio Educación y Vida.*

*La línea de investigación para este proyecto es Tecnologías Actuales y Sociedad, se enmarca en la sub-línea de sistemas de información y como campo temático corresponde al Almacenamiento de Información.*

#### **3.2 TÉCNICAS DE RECOLECCIÓN**

*La técnica utilizada para recolección de información acorde a las características propias del proyecto, fue la observación directa establecida en un trabajo de campo que abarca desde el componente técnico hasta el componente administrativo y de gestión.*

*Las etapas que se siguieron en el proceso de observación fueron las siguientes:*

- *Determinar la situación.*
- *Determinar los objetivos de la observación.*
- *Observar cuidadosa y críticamente.*
- *Analizar e interpretar los datos*
- *Elaborar conclusiones*

*El elemento preponderante en esta investigación que hace parte del proceso de observación y que marca la justificación de este proyecto esta centrado en la necesidad de disminuir los costos económicos de la institución.*

#### **4. DESARROLLO DEL PROYECTO**

*El desarrollo de este proyecto fue ejecutado bajo la guía de un cronograma planteado, el cual permitió el desarrollo eficiente de los objetivos trazados. (Ver Ilustración 2)*

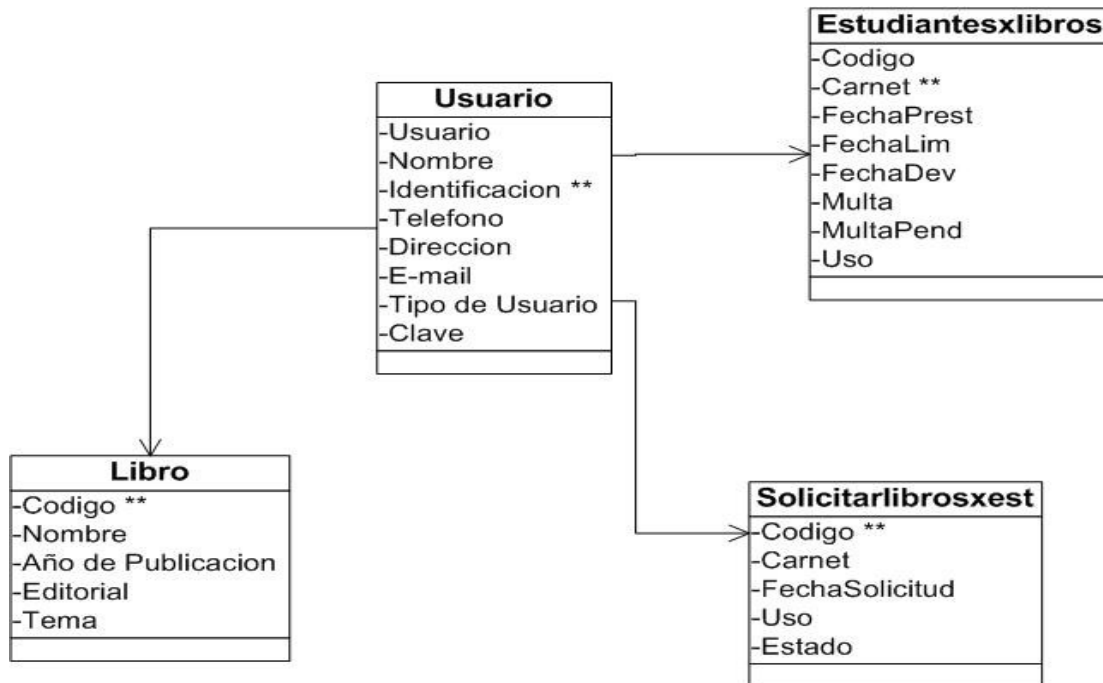
*El diseño para la implementación del Software libre para el manejo de inventario, circulación y préstamo de libros del colegio Educación y Vida, comprende varias etapas, que inician desde el levantamiento de información, el cual, permite conocer el estado actual y los procesos operantes en la biblioteca del colegio; así como, las necesidades a nivel de tecnología informática, en busca de una solución que optimice costos de operación y que asegure la regulación de los procesos propios de la biblioteca.*

*Los agentes vinculados en el proceso a optimizar, son estudiantes, profesores, personal administrativo y recurso financiero. La solución debe asegurar la operación de los servicios actualmente implementados, con restricciones que permitan la administración del sistema y el uso de los beneficiarios del servicio.*

*Los procesos que operan en la biblioteca inician en la inscripción de los usuarios los cuales deben ser discriminados en dos tipos diferentes administrador y usuario, lo que permite el acceso con limitaciones al aplicativo, se hace el ingreso de nuevos ejemplares al inventario, también se ejecuta consulta y solicitud de libros por parte de los usuarios, y dichas solicitudes deben ser enviadas en línea al administrador del aplicativo, lo que permite la entrega del material solicitado, dicho material debe ingresar al inventario cuando se haga la devolución del mismo lo que permitirá un control estricto del inventario y stock de libros.*

*La biblioteca debe tener como mínimo 1 equipo de cómputo, 1 impresora y el personal que los opere. Lo que permite utilizar los equipos de cómputo existentes, dado que funciona sobre un computador convencional (CPU, monitor, teclado y ratón). Teniendo en cuenta que la solución es totalmente escalable, puede operarse en ambiente cliente/ servidor cuando nuevos recursos tecnológicos estén disponibles*

Después de realizar el estudio de la información y definir las necesidades propias del software se realizó la comparación de los sistemas administradores de bases de datos (ver Tabla 1), además se realizó el diseño de las bases de datos, lo cual permitió la realización del diagrama Entidad- relación el cual determina la utilidad de las tablas de la Base de Datos.



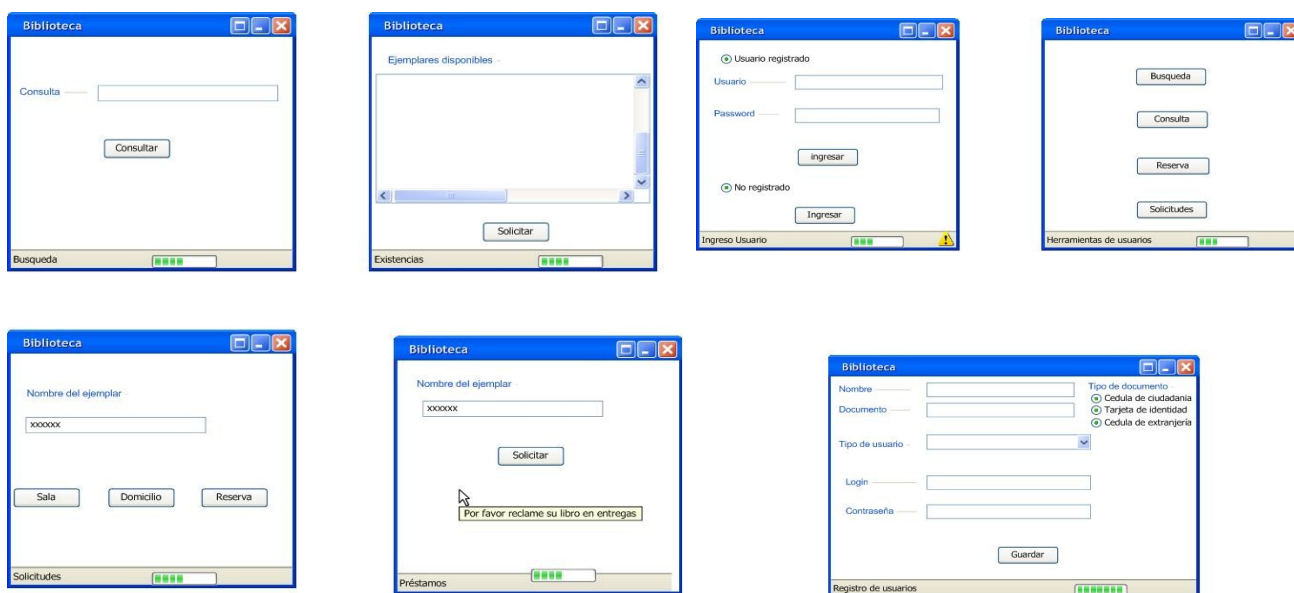
**Ilustración 2. Diagrama Entidad - Relación**

Dentro de las posibilidades de operación de la herramienta estará la de conectar dos puntos de atención entre sí (red punto a punto), manteniendo información actualizada en línea y mejorando tiempos de atención, del mismo modo se convierte en un proceso contingente que asegure la operación continua del proceso.

Luego se definió un diseño preliminar de interfaz de usuario que permitiría el manejo del aplicativo para dos tipos de usuario (Administrador y Usuario), lo cuales tendrían diferentes privilegios.

Cada operación en el software, se visualiza como un proceso de registro de actividades que afianza el control de inventarios, disponibilidades y usuarios; convirtiéndose en una herramienta que arroje indicadores y permita aplicar planes de mejoramiento.

*Después del diseño se concibe una nueva aplicación la cual recibe un nombre de usuario, al que asocia a un código, algunos datos demográficos de registro y una contraseña personal e intransferible. Al objeto usuario se le asocia una lista de solicitudes y unos tiempos de acuerdo al reglamento de la biblioteca. Esta misma información es replicada a un perfil administrador que opera en nombre de la biblioteca. Los tiempos de entrega estipulados, se deben cumplir, de lo contrario se aplican las multas impuestas que serán cargadas debidamente en el sistema. Cuando una multa está vigente, las solicitudes del usuario multado son negadas, una vez la biblioteca reciba la notificación correspondiente, la sanción será levantada.*



**Ilustración 3. Diseño de Interfaz de usuario**

*Para la puesta en marcha del proyecto, se realizará un proceso de instalación y configuración del software y con el personal vinculado al proceso diario de la biblioteca se seguirán tareas de capacitación en la herramienta (aplicado a personal de la biblioteca y a usuarios finales). Este proceso de capacitación se realiza mediante cronograma definido conjuntamente con el personal de la biblioteca y con un acompañamiento por el lapso de 3 días en la operación normal de los usuarios. Así mismo, con un manual de usuario que apoye el proceso.*

*En el proceso de desarrollo de la herramienta se realizó una comparación de los diferentes lenguajes de programación libres (ver Tabla 4), siguiendo la línea de optimización de costos y por robustez, disponibilidad, licenciamiento y dado que*

*cumple con los requerimientos, se determinó como opción más acertada el lenguaje PHP y MySQL.*

## 5. CONCLUSIONES

*Después del análisis de las necesidades del aplicativo a desarrollar, se noto que el aplicativo ha constituido un gran complemento ya que la redacción y construcción de este proyecto ha supuesto dedicar numerosas horas de trabajo en el diseño de las bases de datos, este diseño permitió observar y verificar las necesidades en cuanto a tablas de almacenamiento de datos y su posible relación, además se dedico un período para la concepción y creación de las interfaces previas, lo cual arrojó la necesidad de crear diferentes tipos de usuario con el fin del manejo de tipos de seguridad según el perfil establecido.*

*El resultado ha sido la construcción de una aplicación que corre bajo software libre, con estructura cliente servidor, que permite las solicitudes por parte de los usuarios, y un manejo ágil y eficaz por parte de un administrador.*

*En cuanto a la estructura de navegación hay que destacar la puesta en práctica de un sistema de navegación y un sistema de búsqueda específicos con el fin de facilitar al usuario un uso eficaz de este aplicativo. Así pues podemos concluir que después de los estudio los realizados, del análisis del software actual, del diseño de las bases de datos, la planeación de las interfaces de usuario y de la experiencia de construir el software se llega a la conclusión que la opción mas acertada en el momento de construir una aplicación que responda a las necesidades migración y actualización de software es la utilización de software libre, ya que este disminuye los costos debido a que no cuenta con licenciamiento y además permite la actualización a muy bajo costo.*

*Por tal motivo se da como solución final que el software de Biblioteca creado con la herramienta de PHP, es una muy viable opción para la migración del sistema actual.*

## **6. RECOMENDACIONES**

*Se hace pertinente la realización de capacitaciones a usuarios para familiarizarlos con el nuevo ambiente para reconocer sus ventajas virtudes y operacionalidad.*

*Además es de gran importancia tanto para este aplicativo como para cualquier otro, la realización de copias de seguridad de las bases de datos, ya que esto ayudara a conservar la información en caso de que se presente algún inconveniente. Estas deben realizarse de ser posible en un disco duro diferente al del equipo servidor, semanalmente y conservando todas las copias realizadas.*

## **BIBLIOGRAFÍA**

*ProArgentina(Editor). Industria del software.  
Argentina: El Cid Editor, 2005. p 16.*

*ProArgentina(Editor). Software/ América Latina.  
Argentina: El Cid Editor, 2005. p 3.*

*Gil Rivera, Maria del Carmen. La base de datos: importancia y aplicación en educación.  
México: Red Perfiles Educativos, 2006. p 4.*

*BYERS, Robert, A.  
1988. Introducción a las bases de datos con dBASE III Plus. tr. Luis Hernández  
Yañez,  
México, McGraw-Hill, 365 pp.*

*Cobo, Ángel. PHP y MySQL: tecnologías para el desarrollo de aplicaciones web.  
España: Ediciones Díaz de Santos, 2007. p 23.*

*Wong, Debbie. Oracle8i DBA Backup & Recovery.  
Scottsdale, AZ, USA: Coriolis Group, LLC, The, 2001. p ii.*

*Holzner, Steven. Java.  
Phoenix, AZ, USA: Paraglyph Press, 2001. p xxx.*

## **GLOSARIO**

**Consulta:** Es uno de los procesos principales en cualquier sistema de Bibliotecas ya que este proceso ayuda a los usuarios a encontrar la información deseada,

**Préstamo:** Es el paso por el cual un usuario utiliza el servicio de la biblioteca.

**Sistemas operativos:** Es el programa que controla diferentes procesos en la computadora, uno de los mas importantes es la interpretación de comandos que permiten al usuario comunicarse con el ordenador.

**Intérpretes de instrucciones:** Están basados en texto y exigen que las instrucciones sean tecleadas. Otros son basados en gráficos y permiten al usuario comunicarse señalando y haciendo clic en el icono.

**Software libre:** Es el software con autorización para que cualquiera pueda usarlo, copiarlo y distribuirlo, ya sea con o sin modificaciones, gratuitamente o mediante una retribución. En particular, esto significa que el código fuente debe estar disponible.

**Software GNU:** Software liberado bajo el auspicio del proyecto GNU. El archivo README o el anual del paquete GNU deberían identificar lo que es; además el directorio Software Libre identifica todos los paquetes GNU,

**Copyleft:** Copia permitida y describe un grupo de derechos aplicados a una diversidad de trabajos tales como programas informáticos, arte, cultura y ciencia,

**Software privativo:** Este tipo de software no es libre ni semilibre. Su uso, redistribución o modificación están prohibidos, se requiere una autorización, o esta tan restringido que de hecho no se puede hacer libremente.

**Software de dominio publico:** Este no esta protegido por los derechos de autor, el software libreo no protegido por Copyleft incluye una autorización para distribuir y modificar el software, así como permiso para añadirle restricciones adicionales,,

**Software semilibre:** Este tipo de software no es libre pero incluye autorización para que los particulares lo usen, lo copien, lo distribuyan y lo modifiquen sin beneficios lucrativos

**Freeware:** Tipo de software que se puede distribuir pero no modificar.

**Shareware:** Tipo de software que permite redistribuir copias, pero cada copia utilizada debe pagar un cargo por licencia

**Software comercial:** Software desarrollado que pretende obtener dinero por su utilización.

**Interfaz de usuario:** Es la forma en que el usuario puede comunicarse con la computadora y comprende los puntos de contacto entre el usuario y la computadora

**Lenguaje PHP:** Lenguaje interpretado del lado del servidor, el cual se caracteriza por su potencia, versatilidad, robustez y modularidad.

**Lenguaje JAVA:** este es un lenguaje muy parecido a C/C++, es bastante seguro y su seguridad es configurable, permitiendo restringir el acceso a archivos de una red.

**Bases de datos:** Conocen las ventajas de acumular y cruzar la información sobre un tema con otro tema, están guardan un alto grado de homogeneidad debido a las normas y criterios internacionales creados para la clasificación de todo tipo de documentos.

**Motores de bases de datos:** son un conjunto exhaustivo no redundante de datos estructurados organizados independientemente de su utilización y su implementación en maquinas accesibles en tiempo reales

**Modelo Relacional:** Modelo de base de datos que se emplea con mas frecuencia en la practica, debido a las ventajas que ofrece sobre los demás modelos, es de rápido entendimiento por parte de los usuarios

**PostgresQL:** Sistemas de gran alcance, funciona con todos los sistemas operativos es obediente tiene apoyo para las llaves extranjerias, ensambla , las opiniones, los disparadores y los procedimientos almacenados

**Mysql:** Sistema de administración de bases de datos relacionales rápido, sólido y flexible, es ideal para bases de datos con acceso desde paginas Web dinámicas, y para la creación de transacciones on-line.

**Oracle:** Sistema de gestión de bases de datos relacional, este se considera bastante completo, ya que se destaca si soporte de transacciones, estabilidad, escalabilidad, y es multiplataforma.

**Códigos de Barras:** Representación de una determinada información mediante un conjunto de líneas paralelas verticales de diferente grosor y espaciado, sirve para reconocer rápidamente un artículo de una base de datos,

**EAN:** Sistema de códigos adoptado por mas de 100 países, esta constituido por 13 dígitos y dividido en 4 partes, y el código ISBN que es un identificador único para libros, previsto para uso comercial.

## ANEXOS

*Tabla 1. Comparación de sistemas administradores de bases de datos relacionales*

|                                   | <i>Creador</i>                                  | <i>Última versión estable</i> | <i>Licencia de software</i>       |
|-----------------------------------|-------------------------------------------------|-------------------------------|-----------------------------------|
| <i>Adaptive Server Anywhere</i>   | <i>Sybase/iAnywhere</i>                         | <i>10.0</i>                   | <i>Propietario</i>                |
| <i>Adaptive Server Enterprise</i> | <i>Sybase Inc</i>                               | <i>15.0 (Sep. 2005)</i>       | <i>Propietario</i>                |
| <i>ANTs Data Server</i>           | <i>ANTs Software</i>                            | <i>3.0</i>                    | <i>Propietario</i>                |
| <i>DB2</i>                        | <i>IBM</i>                                      | <i>9</i>                      | <i>Propietario</i>                |
| <i>Firebird</i>                   | <i>Firebird Foundation</i>                      | <i>2</i>                      | <i>Licencia Pública InterBase</i> |
| <i>Informix</i>                   | <i>Informix Software</i>                        | <i>10.0</i>                   | <i>Propietario</i>                |
| <i>HSQldb</i>                     | <i>Hsqldb.Org</i>                               | <i>1.8.0</i>                  | <i>Licencia BSD</i>               |
| <i>Ingres</i>                     | <i>Berkeley University, Computer Associates</i> | <i>2006</i>                   | <i>CA-TOSL</i>                    |
| <i>InterBase</i>                  | <i>Borland</i>                                  | <i>7.5.1</i>                  | <i>Propietario</i>                |
| <i>SapDB</i>                      | <i>SAP AG</i>                                   | <i>7.4</i>                    | <i>GPL con drivers LGPL</i>       |
| <i>MaxDB</i>                      | <i>MySQL AB, SAP AG</i>                         | <i>7.5</i>                    | <i>GPL o propietario</i>          |
| <i>Microsoft SQL Server</i>       | <i>Microsoft</i>                                | <i>9.00.2047 (2005 SP1)</i>   | <i>Propietario</i>                |
| <i>MySQL</i>                      | <i>MySQL AB</i>                                 | <i>5.0</i>                    | <i>GPL o propietario</i>          |
| <i>Oracle</i>                     | <i>Oracle Corporation</i>                       | <i>10g Release 2</i>          | <i>Propietario</i>                |
| <i>PostgreSQL</i>                 | <i>PostgreSQL Global Development Group</i>      | <i>8.2.3</i>                  | <i>Licencia BSD</i>               |
| <i>SmallSQL</i>                   | <i>SmallSQL</i>                                 | <i>0.12</i>                   | <i>LGPL</i>                       |
| <i>SQLite</i>                     | <i>D. Richard Hipp</i>                          | <i>3.1.3</i>                  | <i>Dominio público</i>            |
|                                   | <i>Creador</i>                                  | <i>Última versión estable</i> | <i>Licencia de software</i>       |

Tabla 2. Soporte del sistema operativo

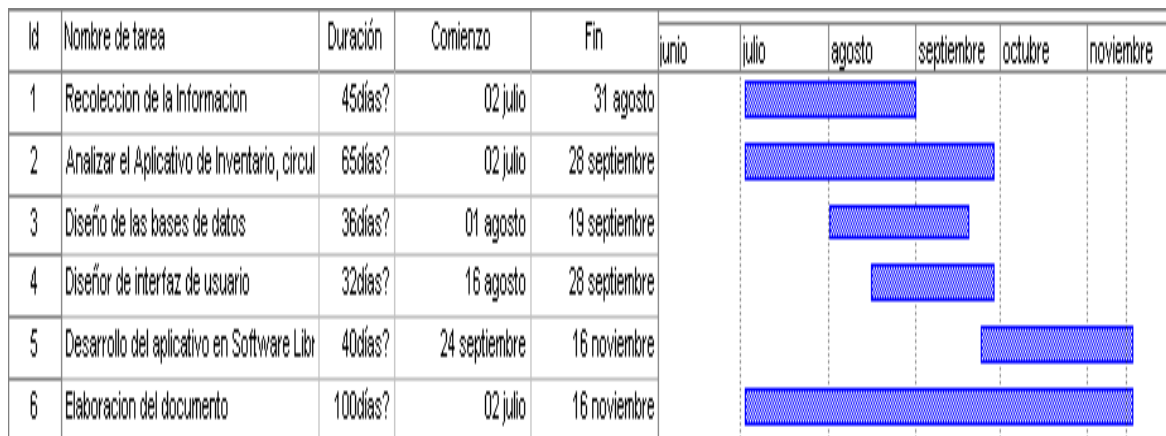
|                                   | <b>Windows</b> | <b>Mac OS X</b> | <b>Linux</b> | <b>BSD</b> | <b>Unix</b>             | <b>z/OS</b>  |
|-----------------------------------|----------------|-----------------|--------------|------------|-------------------------|--------------|
| <i>Adaptive Server Enterprise</i> | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>No</i>    |
| <i>ANTs Data Server</i>           | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>?</i>     |
| <i>DB2</i>                        | <i>Sí</i>      | <i>No</i>       | <i>Sí</i>    | <i>No</i>  | <i>Sí</i>               | <i>Sí</i>    |
| <i>Firebird</i>                   | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>Quizá</i> |
| <i>HSQLDB</i>                     | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>Sí</i>    |
| <i>Informix</i>                   | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>No</i>    |
| <i>Ingres</i>                     | <i>Sí</i>      | <i>?</i>        | <i>Sí</i>    | <i>?</i>   | <i>Sí</i>               | <i>Quizá</i> |
| <i>InterBase</i>                  | <i>Sí</i>      | <i>No</i>       | <i>Sí</i>    | <i>No</i>  | <i>Sí<br/>(Solaris)</i> | <i>No</i>    |
| <i>SapDB</i>                      | <i>Sí</i>      | <i>No</i>       | <i>Sí</i>    | <i>No</i>  | <i>Sí</i>               | <i>?</i>     |
| <i>MaxDB</i>                      | <i>Sí</i>      | <i>No</i>       | <i>Sí</i>    | <i>No</i>  | <i>Sí</i>               | <i>?</i>     |
| <i>Microsoft SQL Server</i>       | <i>Sí</i>      | <i>No</i>       | <i>No</i>    | <i>No</i>  | <i>No</i>               | <i>No</i>    |
| <i>MySQL</i>                      | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>Quizá</i> |
| <i>Oracle</i>                     | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>No</i>  | <i>Sí</i>               | <i>Sí</i>    |
| <i>PostgreSQL</i>                 | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>No</i>    |
| <i>SmallSQL</i>                   | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>Sí</i>    |
| <i>SQLite</i>                     | <i>Sí</i>      | <i>Sí</i>       | <i>Sí</i>    | <i>Sí</i>  | <i>Sí</i>               | <i>Quizá</i> |
|                                   | <b>Windows</b> | <b>Mac OS X</b> | <b>Linux</b> | <b>BSD</b> | <b>Unix</b>             | <b>z/OS</b>  |

*Tabla 3. Características fundamentales*

|                                   | <b><i>ACID</i></b> | <b><i>Integridad<br/>referencial</i></b> | <b><i>Transacciones</i></b> | <b><i>Unicode</i></b> |
|-----------------------------------|--------------------|------------------------------------------|-----------------------------|-----------------------|
| <i>Adaptive Server Enterprise</i> | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>ANTs Data Server</i>           | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>DB2</i>                        | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>Firebird</i>                   | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>HSQldb</i>                     | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>Informix</i>                   | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>Ingres</i>                     | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>InterBase</i>                  | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>SapDB</i>                      | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>MaxDB</i>                      | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>Microsoft SQL Server</i>       | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>MySQL</i>                      | <i>Depende</i>     | <i>Depende</i>                           | <i>Depende</i>              | <i>Sí</i>             |
| <i>Oracle</i>                     | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>PostgreSQL</i>                 | <i>Sí</i>          | <i>Sí</i>                                | <i>Sí</i>                   | <i>Sí</i>             |
| <i>SQLite</i>                     | <i>Sí</i>          | <i>No</i>                                | <i>Básico</i>               | <i>Sí</i>             |
|                                   | <b><i>ACID</i></b> | <b><i>Integridad<br/>referencial</i></b> | <b><i>Transacciones</i></b> | <b><i>Unicode</i></b> |

Tabla 4. Comparación entre software libre y software privativo

| TAREAS EJECUTADAS             | SOFTWARE PRIVATIVO                 | SOFTWARE LIBRE                                                                                                                                                                                                  |
|-------------------------------|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lenguajes para Desarrollo Web | PHP para Windows                   | <u>PHP.</u>                                                                                                                                                                                                     |
| Lenguajes para desarrollo web | Perl para Windows                  | <u>Perl.</u>                                                                                                                                                                                                    |
| Lenguajes para desarrollo web | ASP                                | ASP módulos para Apache.                                                                                                                                                                                        |
| Motores de Bases de Datos     | MS SQL, MySQL para Windows         | 1) <u>Sybase Adaptive Server Enterprise.</u> [Prop]<br>2) <u>PostgreSQL.</u> Las mas avanzada base de datos.<br>3) <u>MySQL.</u> Las mas popular base de datos abierta.<br>4) <u>mSQL.</u><br>5) <u>SAP DB.</u> |
| Motores de Bases de Datos     | IBM DB2                            | <u>IBM DB2 para Linux.</u> [Prop]                                                                                                                                                                               |
| Motores de Bases de Datos     | Oracle para Windows                | 1) <u>Oracle.</u> [Prop]<br>2) <u>PostgreSQL.</u><br>3) <u>Linter.</u> (cyrillic)                                                                                                                               |
| Motores de Bases de Datos     | Informix                           | Informix.                                                                                                                                                                                                       |
| Motores de Bases de Datos     | Borland Interbase, <u>FireBird</u> | <u>FireBird.</u>                                                                                                                                                                                                |



**Ilustración 4. Cronograma de desarrollo del proyecto**

### 3. METODOLOGÍA

#### 3.1. ENFOQUE DE LA INVESTIGACIÓN

Este proyecto constituye una investigación de tipo empírico – analítico ya que, analizando el proceso de diligenciamiento de reportes en la actualidad, se ha determinado que existen varios puntos de mejora como lo son el control, la seguridad y la automatización del mismo. Estas mejoras se lograrán mediante la realización de una interfaz que permita diligenciar el formato de reporte de operación sospechosa, de esta manera se minimiza el tiempo del proceso gracias a la eliminación de la impresión y el diligenciamiento manual asegurando así una mayor efectividad en el tiempo de respuesta gracias a la automatización del proceso de diligenciamiento; además, mediante la encriptación de la información, esta interfaz da a la información el nivel de seguridad adecuado garantizando así un mayor control en el proceso.

La línea de investigación para este proyecto es Tecnologías Actuales y Sociedad, se enmarca en la sub-línea de sistemas de información y como campo temático corresponde al Almacenamiento de Información.

### **3.2 TECNICAS DE RECOLECCION**

La técnica utilizada para la recolección de información fue la observación. Para el proyecto se considera observación, al uso sistemático de los sentidos orientados a la captación de la realidad que está en el estudio. Es una técnica antigua: a través de los sentidos, el hombre capta la realidad que lo rodea, que luego organiza intelectualmente. Durante innumerables observaciones sistemáticamente repetidas. El uso de los sentidos es una fuente inagotable de datos que, tanto para la actividad científica como para la vida práctica, resulta de inestimable valor. En la aplicación de esta técnica, se registra lo observado, mas no se interroga a los individuos involucrados en la operativa de la operación; es decir, no hace preguntas, orales o escrita, que le permitan obtener los datos necesarios para el estudio del problema. La observación tiene la ventaja de facilitar la obtención de datos lo más próximos a como éstos ocurren en la realidad.