



UNIVERSIDAD TÉCNICA PARTICULAR DE LOJA
La Universidad Católica de Loja

ÁREA TÉCNICA

TÍTULO DE INGENIERO EN INFORMÁTICA

**Análisis y evaluación de soluciones
de software libre para gestión de
bibliotecas universitarias en Ecuador**

TRABAJO DE TITULACIÓN.

AUTOR: Vela Vargas, Gil Eduardo

DIRECTOR: Elizalde Solano, René Rolando, Ing.

CENTRO UNIVERSITARIO QUITO TUMBACO

2017

APROBACIÓN DEL DIRECTOR DEL TRABAJO DE TITULACIÓN

Ingeniero

René Rolando Elizalde Solano

DOCENTE DE LA TITULACIÓN

De mi consideración:

El presente trabajo de titulación Análisis y evaluación de soluciones de software libre para gestión de bibliotecas universitarias del Ecuador realizado por Gil Eduardo Vela Vargas ha sido orientado y revisado durante su ejecución, por cuanto se aprueba la presentación del mismo.

Loja, febrero de 2017

f. _____

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

Yo Gil Eduardo Vela Vargas declaro ser autor del presente trabajo de titulación: Análisis y evaluación de soluciones de software libre para gestión de bibliotecas universitarias en Ecuador, de la Titulación de Ingeniero en Informática, siendo el Ing. René Elizalde Solano director del presente trabajo; y eximo expresamente a la Universidad Técnica Particular de Loja y a sus representantes legales de posibles reclamos o acciones legales. Además certifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad.

Adicionalmente declaro conocer y aceptar la disposición del Art. 88 del Estatuto Orgánico de la Universidad Técnica Particular de Loja que en su parte pertinente textualmente dice: “Forman parte del patrimonio de la Universidad la propiedad intelectual de investigaciones, trabajos científicos o técnicos y tesis de grado o trabajos de titulación que se realicen con el apoyo financiero, académico o institucional (operativo) de la Universidad”

f. _____

Autor: Gil Eduardo Vela Vargas

Cédula: 1707318851

DEDICATORIA

A todos los que invariablemente creyeron en mi potencial para alcanzar esta meta y me acompañaron y apoyaron a lo largo de estos años.

A Belén, Paola y Gabriel por su paciencia y motivación.

A mis padres por su constante apoyo y preocupación.

A mis hermanos Miguel Angel(+), Ana Cristina y Gandhi por haber recorrido juntos gran parte de la vida y estar juntos en las buenas y en las malas.

Eduardo Vela

AGRADECIMIENTO

A la Universidad Técnica Particular de Loja, a todos mis profesores y particularmente al personal de la Sede Quito Tumbaco por su apoyo a lo largo de esta jornada pedagógica que me ha llevado a redescubrir el gusto por el conocimiento y el aprendizaje constante.

INDICE DE CONTENIDOS

CARATULA.....	i
APROBACIÓN DEL DIRECTOR DEL TRABAJO DE FIN DE TITULACIÓN	ii
DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	iii
DEDICATORÍA	iv
AGRADECIMIENTO.....	v
ÍNDICE DE CONTENIDOS	vi
ÍNDICE DE TABLAS.....	xii
ÍNDICE DE ILUSTRACIONES	xiv
Resumen	1
Abstract	2
Introducción.....	3
Objetivos	4
Objetivo General.....	4
Objetivos Específicos	4
Resultados esperados.....	4
Capítulo 1	
Marco Teórico de la Investigación	5
1.1 El Software libre.....	6
1.1.1 Historia.....	6
1.1.2 Copyleft.....	6
1.1.3 Las 4 Libertades.	7
1.1.4 Evolución.	8
1.2 Situación actual	8
1.2.1 Importancia del uso de licencias de software libre.	9
1.3 Otros tipos de licencias.....	9
1.4 El Software libre en Ecuador	10
1.5 Modelos de evaluación de calidad del software	11
1.6 Importancia de la evaluación de software libre.....	11
1.6.1 Evaluación de software libre.....	12
1.6.2 Particularidades de la evaluación de software libre.	12

Capítulo 2 Metodologías de evaluación de software libre	15
2.1 Inicios de las metodologías de evaluación	16
2.1.1 IRCA.	16
2.1.1.1 Wheeler.	16
2.1.1.2 Mangalan.....	17
2.1.2 QSOS.	19
2.1.3 Qualipso OMM.....	19
2.1.4 Van den Berg	21
2.1.5 Evaluación a través de Análisis de Proyectos	23
2.1.5.1 Requisitos funcionales.....	23
2.1.5.2 Requisitos técnicos.....	23
2.1.5.3 Requisitos organizacionales.	23
2.1.5.4 Requerimientos legales	23
2.1.5.5 Requisitos económicos.....	24
2.1.5.6 Requerimientos políticos.	24
2.1.6 OpenBRR	24
2.1.7 Modelo de Madurez Smals OSS	26
2.2 Análisis comparativo de las metodologías.....	27
2.3 Elección del modelo a utilizar para la evaluación	30
Capítulo 3	
Caso de estudio: Software libre para bibliotecas	
universitarias en Ecuador	31
3.1 Introducción	33
3.2 Selección	35
3.2.1 Criterios de selección	35
3.2.2 Métricas cualitativas y cuantitativas.....	37
3.2.3 Lista de candidatos	38
3.2.4 Eliminación por aspectos.....	39
3.2.5 Valoración de los candidatos seleccionados	42
3.2.5.6.1 Peso específico de criterios de selección.....	42
3.2.5.1 Valoración de Funcionalidad	43
3.2.5.1.1 Arquitectura	43

3.2.5.1.2 Licencia	43
3.2.5.1.3 Tecnología	43
3.2.5.1.4 Estándares	43
3.2.5.1.5 Módulos	44
3.2.5.2 Valoración de Comunidad	46
3.2.5.3 Valoración de Liberación de Versiones	47
3.2.5.4 Valoración de Longevidad del Proyecto	47
3.2.6 Resultados de la selección	48
Capítulo 4	
Caso de estudio parte 2: Evaluación Koha y PMB.....	50
4.1 Introducción	51
4.1.1 KOHA.....	51
4.1.2 PMB	52
4.2 Comunidad	53
4.2.1 KOHA.....	53
4.2.2 PMB	54
4.2.3 Calificaciones.....	55
4.3 Liberación de versiones.....	56
4.3.1 Koha	56
4.3.2 PMB.....	56
4.3.3 Calificaciones.....	56
4.4 Longevidad	57
4.4.1 edad y version	57
4.4.1.1 Koha	57
4.4.1.2 PMB.....	57
4.4.2 Calificaciones.....	57
4.4.3 Nuevas tecnologías	57
4.4.3.1 Koha	57
4.4.3.2 PMB.....	58
4.4.3.3 Calificaciones	58
4.5 Licencias.....	58
4.5.1 Koha	58

4.5.2 PMB	58
4.5.3 Calificaciones.....	59
4.6 Soporte	59
4.6.1 Comunidad	59
4.6.1.1 Koha	59
4.6.1.2 PMB.....	59
4.6.2 Soporte pagado	59
4.6.2.1 Koha	59
4.6.2.2 PMB.....	60
4.6.2.3 Calificaciones	61
4.7 Documentación.....	61
4.7.1 Koha	61
4.7.2 PMB	62
4.7.3 Calificaciones.....	62
4.8 Seguridad	62
4.8.1 Koha	62
4.8.2 PMB.....	63
4.8.3 Calificaciones.....	63
4.9 Funcionalidad	64
4.9.1 Koha	64
4.9.1.1 Módulo de Circulación.....	64
4.9.1.2 Módulo de Usuarios.....	64
4.9.1.3 Módulo de Búsqueda Avanzada.....	64
4.9.1.4 Módulo de Listas	64
4.9.1.5 Módulo de Catalogación.....	65
4.9.1.6 Módulo de Autoridades.....	65
4.9.1.7 Módulo de Publicaciones Periódicas.....	65
4.9.1.8 Módulo de Adquisiciones.....	65
4.9.1.9 Módulo de informes	65
4.9.1.10 Módulo Herramientas	66
4.9.1.10.1 Usuarios y circulación.....	66
4.9.1.10.2 Catálogo	66

4.9.1.10.3 Herramientas adicionales	66
4.9.1.10.4 Módulo de Administración de Koha	67
4.9.1.10.5 Módulo Acerca de Koha	67
4.9.2 PMB	67
4.9.2.1 Módulo de Circulación	67
4.9.2.1.1 Cestas	67
4.9.2.1.2 Reservas , Sugerencias y Caducados	68
4.9.2.2 Módulo de Catálogo	68
4.9.2.3 Módulo de Autoridades.....	68
4.9.2.4 Módulo de Informes.....	68
4.9.2.5 Módulo de Diseminación Selectiva de la Información (DSI).....	68
4.9.2.6 Módulo de Administración	69
4.9.3 Calificaciones.....	69
4.10 Integración.....	70
4.10.1 Modularidad.....	70
4.10.2 Estándares.....	70
4.10.3 Compatibilidad con otras aplicaciones	71
4.10.3.1 Requerimientos del software.....	71
4.10.4 Calificaciones.....	72
4.11 Origen y objetivos	72
4.11.1 Calificaciones.....	73
4.12 Resultados.....	73
4.13 Conclusiones del caso de estudio	74
4.14 Análisis de validez de los resultados obtenidos	75
4.14.1 Prueba de funcionalidad en importación de registros y rendimiento general.....	75
4.14.1.1 Configuración de servidor de pruebas	76
4.14.1.2 Instalación y configuración de los paquetes de software	77
4.14.1.2.1 Instalación de KOHA	77
4.14.1.3 Instalación PMB.....	79
4.14.1.4 Migración de registros desde SIABUC 9	81
4.14.1.4.1 KOHA.....	81

4.14.1.4.2 PMB.....	82
4.14.1.5 Formato MARC21 / ISO 2709 para intercambio de información bibliográfica	84
4.14.1.6 Comparación de resultados de búsquedas en Koha y PMB.....	86
4.14.2 Contratación de resultados con evaluaciones similares.....	87
Capítulo 5	
Conclusiones generales	88
Bibliografía.....	92
Anexo A	
Glosario Terminológico	103
Anexo B	
Encuesta de bibliotecas universitarias	106

ÍNDICE DE TABLAS

Tabla No.1: Tipos de licencias	10
Tabla No.2: Comparación entre software libre y propietario.....	13
Tabla No.3: Métricas cualitativas	18
Tabla No.4: Métricas cuantitativas.....	18
Tabla No.5: Nivel básico.....	20
Tabla No.6: Nivel intermedio.....	20
Tabla No.7: Nivel avanzado.....	21
Tabla No.8: Items para tomar en cuenta en la evaluación OpenBRR.....	25
Tabla No.9: Comparación de características de metodologías de evaluación.....	28
Tabla No.10: Criterios de evaluación metodología van den Berg	36
Tabla No.11: Métricas cualitativas y cuantitativas	37
Tabla No.12: Lista de candidatos	39
Tabla No.13: Lista de eliminación/ paso a ronda final	40
Tabla No.14: Lista de candidatos seleccionados	42
Tabla No.15: Peso porcentual de criterios de selección.....	42
Tabla No.16: Valoración de Funcionalidad	44
Tabla No.17: Valoración de Comunidad	46
Tabla No.18: Valoración Versiones.....	47
Tabla No.19: Valoración de Longevidad	47
Tabla No.20: Resumen de resultados	48
Tabla No.21: Posiciones en valoración final.....	48
Tabla No.22: Resumen calificaciones comunidad.....	55
Tabla No.23: Resumen calificaciones liberación de versiones.....	56
Tabla No.24: Resumen calificaciones longevidad	57
Tabla No.25: Resumen calificaciones nuevas tecnologías	58
Tabla No.26: Resumen calificaciones licencias.....	59
Tabla No.27: Resumen calificaciones soporte.....	61

Tabla No.28: Resumen calificaciones documentación	62
Tabla No.29: Resumen calificaciones seguridad	63
Tabla No.30: Resumen calificaciones funcionalidad	69
Tabla No.31: Comparativa estándares Koha y PMB	70
Tabla No.32: Resumen calificaciones integración	72
Tabla No.33: Resumen calificaciones origen y objetivos.....	73
Tabla No.34: Resultados de evaluación de Koha y PMB	74
Tabla No.35: Visualización de registro MARC21	85

ÍNDICE DE ILUSTRACIONES

Figura 1 Logotipo de Copyleft.	7
Figura 2 Gráfico explicativo metodología a utilizar en la evaluación.....	32
Figura 3 Logotipo de Koha	51
Figura 4 Logotipo de PMB.....	52
Figura 5 Importación de registros bibliográficos desde el software SIABUC 9	76
Figura 6 Servidor configurado con 2 máquinas virtuales en VirtualBox	77
Figura 7 Instalación y configuración de Koha.....	78
Figura 8 Interface de administración de Koha instalada	79
Figura 9 Instalación de PMB.....	80
Figura 10 Interface administrativa de PMB funcionando.....	80
Figura 11 Preparación para importación de registros a Koha	81
Figura 12 Confirmación de importación de registros a Koha.....	82
Figura 13 Importación de registros a PMB	83
Figura 14 Registros importados disponibles en PMB.....	84
Figura 15 Pantallas de resultado de búsqueda de Koha (superior) y PMB (inferior) .	86
Figura 16 Resultados de evaluación de Muller.....	87

RESUMEN

El impacto y adopción que han tenido las tecnologías de la información en el desarrollo de todo tipo de actividades humanas se ha visto incrementado por el crecimiento en el apareamiento del software libre en los últimos años en todo el mundo y en particular en los países en vías de desarrollo como el nuestro.

La evaluación y selección de este tipo de software para su configuración e implementación se vuelve por tanto en una tarea clave en las responsabilidades del personal y departamento de informática de todo tipo de instituciones, por lo que el estudio y uso de una o varias metodologías que se orienten a este fin son de particular importancia.

En el presente trabajo se busca analizar el contexto de la aparición del movimiento de software libre así como sus particularidades y las principales metodologías propuestas para la evaluación de las mismas.

Posteriormente se desarrolla la implementación metodológica que permita realizar una evaluación de software para la gestión de bibliotecas orientada hacia su uso en las bibliotecas universitarias en nuestro país.

PALABRAS CLAVES: software libre, evaluación, universidad, biblioteca, metodología

ABSTRACT

Information technologies have been a tremendous impact and adoption in recent years in every kind of human activities. This trend has increased greatly with the appearance of open source software projects worldwide, specially in developing countries like Ecuador.

Selection and evaluation of this software prior its configuration and implementatio becomes an important task por IT departments and staff everywhere. An adequate study and use of one or many methodologies towards this objective then becomes a key assignment for IT professionals.

This document tries to study the emergence of the open source software movement, its unique characteristics and the main methodologies developed to evaluate them.

Finally a methodology implementation will be made to evaluate open source software for library management for a potencial use in Ecuador's universities.

KEYWORDS: open source software, evaluation, university, library, methodology

INTRODUCCIÓN

En los últimos años la fuerza que ha tomado la adopción de software de acceso libre (free/ libre open source software o FLOSS) ha generado un cambio profundo en la manera como las personas y sociedades se relacionan con la informática y la tecnología en general.

El concepto de software libre es un modelo de desarrollo y colaboración que sobre todo promueve el libre acceso al conocimiento, en este caso específico a las aplicaciones software, su uso, personalización, distribución y modificación.

La cada vez mayor aparición y adopción de soluciones de esta clase de software implica nuevas formas de relacionarse e involucrarse con las tecnologías de la información, con particularidades distintas a las existentes anteriormente hacia el modelo de uso de software propietario, en especial en los métodos de evaluación y selección, análisis, capacitación, implementación y soporte.

El énfasis especial de este trabajo enfocado en la comunidad bibliotecaria universitaria es el poner en contexto el gran impacto que puede tener en esta el aprovechamiento de software libre para la gestión de sus contenidos, tomando en cuenta que existe un paralelismo entre los objetivos de las bibliotecas como un vehículo inclusivo para el acceso al conocimiento y los movimientos de software libre y acceso abierto, por lo que resulta natural apoyarse en herramientas de software libre y Open Access para mejorar su desempeño y redefinir su modelo de servicios.

Al igual que la educación universitaria va evolucionando gracias a nuevas herramientas, las bibliotecas universitarias deben ser parte de este cambio, siendo esta la razón principal de este trabajo de investigación, primeramente para realizar un estudio de metodologías de evaluación de software libre en general y finalmente enfocado como una guía de referencia de las nuevas tecnologías de software libre para el desarrollo de nuevos productos y servicios bibliotecarios dentro de las universidades.

OBJETIVOS

Objetivo General

Desarrollar un modelo de evaluación de software libre de gestión integral de bibliotecas para que sirva de referencia para una adecuada selección de software para centros universitarios basado en metodologías y estándares de evaluación de software, generando un análisis comparativo, técnico y objetivo enfocado en las necesidades de las bibliotecas universitarias del país de acuerdo a los requisitos e infraestructura disponible.

Objetivos Específicos

Detallar un resumen de la historia de la evolución, particularidades e importancia del movimiento de software libre.

Investigar y reseñar las características de las principales y más recientes metodologías de evaluación de software libre.

Seleccionar la metodología más adecuada para la evaluación de software de software libre para gestión de bibliotecas universitarias en el Ecuador y desarrollar la misma.

Resultados esperados

Investigar, seleccionar y adaptar una metodología de evaluación de software libre para gestión integral de bibliotecas universitarias.

Generar un análisis cualitativo y cuantitativo de software libre para gestión integral de bibliotecas que detalle características relevantes y adecuación al contexto universitario ecuatoriano.

Disponer de una guía de evaluación que sirva como apoyo en la toma de decisiones para los responsables de la gestión de bibliotecas universitarias sobre las distintas características que las aplicaciones de software libre de gestión bibliotecaria ofrecen y el nivel en que las mismas satisfacen requerimientos de calidad.

CAPÍTULO 1
MARCO TEÓRICO DE LA INVESTIGACIÓN

1.1 El Software libre

1.1.1 Historia.

El concepto de software libre empezó como un movimiento formal a partir de 1971 por parte de Richard Stallman, programador de los Laboratorios de Inteligencia Artificial del MIT (Massachusetts Institute of Technology), en contraposición a la tendencia de las empresas desarrolladoras de software de “encerrar” o bloquear el acceso al código del software que producen, por lo que sus usuarios no pueden realizar cambios o mejoras al mismo y dependen del creador del mismo para esas modificaciones.

Este concepto privilegia el acceso abierto, el desarrollo colaborativo, por lo que Stallman en 1985 funda la FSF (Free Software Foundation) con el objetivo de crear un sistema operativo libre similar a UNIX que se denominó GNU (GNU is Not Unix, GNU No es Unix) pero también fue una iniciativa política, social y ética que se ha ido difundiendo y ganando adeptos en todo el mundo a lo largo del tiempo.

Posteriormente este proyecto se unió a la iniciativa de Linus Torvalds de la Universidad de Helsinki (Finlandia) que desarrolló un kernel sobre los paquetes GNU denominado GNU/Linux en 1992 con un licencia pública GNU (Zurita Sánchez, 2011).

1.1.2 Copyleft.

Uno de los conceptos más importantes desarrollados por la FSF es el de licencia Copyleft que, a través del uso de los instrumentos legales de propiedad intelectual vigentes autorizar el uso, distribución y modificación de los trabajos, en contraposición al Copyright que los prohíbe.

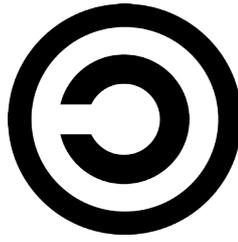


Figura 1 Logotipo de Copyleft.

Fuente: <http://en.wikipedia.org/wiki/Copyleft>

1.1.3 Las 4 Libertades.

El software libre ideológicamente promueve 4 libertades fundamentales y que (Stallman, 2004) detalla:

- Libertad 0. La libertad para ejecutar el programa sea cual sea nuestro propósito.
- Libertad 1: La libertad para estudiar el funcionamiento del programa y adaptarlo a nuestras necesidades.
- Libertad 2: La libertad para redistribuir copias y ayudar así a nuestro vecino.
- Libertad 3: La libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad.

Esta declaración de principios en los que se basa el software libre que busca el libre acceso a la información existente en los sistemas de software y coincide en gran parte con los objetivos de los gobiernos y la sociedad de dedicar inversiones materiales y humanas para que el conocimiento se haga un bien social o se democratice, contribuyendo a la libertad tecnológica y a la inteligencia colectiva, área en la cual las bibliotecas públicas y universitarias deben considerarse como vehículos de inclusión digital (Moreiro González, Rodríguez-Barquin, Martul, & Pinto, 2006).

1.1.4 Evolución.

Desde su aparición hasta la actualidad, y en especial debido a la aparición y crecimiento exponencial de Internet, actualmente el software libre se encuentra presente en casi la totalidad de áreas de la informática moderna, como ejemplo el 89,40% de las más grandes supercomputadoras del mundo utilizan alguna versión del sistema operativo Linux (Internet archive, 2009), al igual que los sistemas informáticos de la Estación Espacial Internacional ISS (ComputerWeekly, 2013).

Como otros ejemplos más comunes, (Wheeler, 2014) nos presenta los siguientes datos:

- Apache, el servidor web más usado en el mundo es software FLOSS (Free Libre Open Source Software).
- Android, basado en Linux es el sistema operativo dominante en teléfonos inteligentes y tablets en todo el mundo.
- En 2006, el 83% de los servidores web públicos funcionan con GNU Linux en contraste con un 23% de Windows, y la tendencia continúa favoreciendo a Linux.
- Desde 2012 la mitad de las aplicaciones de misión crítica corren sobre plataformas Linux.

1.2 Situación actual

Como (Goth, 2005) manifiesta

“Ejecutivos y analistas de la industria de tecnología a menudo no están de acuerdo sobre el significado de un determinado desarrollo o tendencia. Sin embargo, el sentimiento de que la infraestructura de soporte de desarrollo de software libre se está convirtiendo verdaderamente en una ‘fuerza industrial’ es casi unánime ”

1.2.1 Importancia del uso de licencias de software libre.

El uso de licencias permisivas se está acrecentando en proyectos de software, pero también existe muchos casos de desarrolladores que liberan sus códigos sin ninguna licencia en absoluto. Este comportamiento se debe en parte a la cantidad y complejidad de las licencias abiertas que existen en la actualidad.

Pero en el contexto actual en el que el software se está enfocando en ser entregado como un servicio más que implementado, un potencial conflicto se puede dar debido a que gran parte de proyectos de software libre no posee ninguna licencia y que, por defecto, este se considera dentro de las leyes de copyright, lo cual deja abierta a que otras personas se adueñen y reclamen derechos sobre este, potencialmente dejando al autor o autores del mismo con pocos elementos legales con los que defender sus creaciones. (Vaughan-Nichols, 2014)

1.3 Otros tipos de licencias

Existen gran cantidad de licencias de software tanto propietarias como abiertas, por lo que la FSF ha detallado un listado de estas y su compatibilidad o no con el GPL. Entre las licencias de software más conocidas podemos nombrar:

Tabla No.1: Tipos de licencias

Licencias compatibles con GPL	Licencias incompatibles con GPL	Licencias no libres
GNU versión 2 y 3	Licencia Apache versión 1.1	Sin licencia
LGPL 3 y 2.1	BitTorrent Open Source License	Apple Public Source License APSL
Licencia Apache versión 2.0	Licencia BSD original	versión 1.x
Licencia Artística 2.0	EUPL versión 1.1	Licencia Pública AT&T
Licencia BSD Modificada	Licencia pública IBM versión 1.0	The JSON License
CC0	Microsoft Public License	Microsoft's Shared Source CLI, C#,
CeCILL versión 2	Microsoft Reciprocal License	y Jscript License
Licencia Free BSD		NASA Open Source Agreement
Intel Open Source License		Open Public License
Licencia Pública de Mozilla MPL 2		
W3C		
Dominio Público		
Affero AGPL versión 3		

Fuente: (Free Software Foundation, 2015)

1.4 El Software libre en Ecuador

La adopción de software libre en el Ecuador ha ido creciendo de forma acelerada. Incluso se ha creado desde 2014 la Asociación de Software Libre de Ecuador que busca la difusión de su uso, filosofía, valores fundamentales y realizar capacitaciones en el país sobre este movimiento y que acoge como miembros a particulares y personas jurídicas.

A lo largo de su existencia, esta asociación detalla algunos de los hitos más importantes del Software Libre en el país tales como (ASLE, 2014):

- En legislación el Decreto 1024 del año 2008 que coloca como política pública la utilización de Software Libre.
- Igualmente este se menciona en la Ley Orgánica de Educación Superior del 2010, así como en su Reglamento General de 2011.
- El uso de Software Libre se contempla en el Código Ingenios actualmente en debate en

la Asamblea Nacional.

- CEDIA, el Consorcio Ecuatoriano para el Desarrollo de Internet Avanzado ofrece un repositorio de tecnologías libres a través de un espejo público.

1.5 Modelos de evaluación de calidad del software

Las distintas definiciones de concepto de calidad dentro de la industria de software esencialmente definen a esta como la adecuación del producto para cumplir con las expectativas del cliente (Muñoz, Velthuis, & Rubia, 2010).

Justamente el realizar una adecuada evaluación de calidad del software libre es fundamental para determinar que este se encuentre apto para su uso en ambientes de trabajo que han estado acostumbrados a utilizar soluciones de software propietario COTS (Commercial off the shelf).

1.6 Importancia de la evaluación de software libre

Realizar una adecuada evaluación de las alternativas de software libre es cada vez más importante para los responsables de TI en todo tipo de instituciones para un correcto aprovechamiento de los recursos, el talento humano y el tiempo. Como acertadamente (Crum, 2010) lo menciona:

When you're evaluating open sourced software you have to remember that there will always be two perspectives you have to keep in mind. If you're considering it from an IT/Management perspective, if you look hard enough, there will often be a program that meets your platform requirements, is easy to install, and is free. Doing your homework also means evaluating them from the end-user perspective. It's easy to say "they're just going to have to deal with it..." but this more often than not leads to a decrease

in productivity in the least, and at worse a full edged mutiny. Few things are more frustrating than pitching a change and then having to go back to the drawing board when what seemed like a good idea fails and requires reverting back to something you decided was worth leaving in the first place. A little expansion on the evaluation will usually lead to much less trial and error, and ultimately a better fit in the long run.

Al evaluar software libre se debe recordar que siempre habrán dos perspectivas a tomar en cuenta. Si lo consideramos desde una perspectiva TI/Administrativa, si se busca con suficiente empeño, frecuentemente existe un programa que cumple con los requerimientos de plataforma, es fácil de instalar y libre. Pero hacer correctamente la tarea significa evaluar este programa desde el punto de vista del usuario final. Es fácil decir "solo tienen que acostumbrarse..." pero esto más temprano de tarde lleva al menos a una pérdida de productividad, y en el peor de los casos a un motín general. Pocas cosas son más frustrantes que implementar un cambio y luego tener que regresar al inicio cuando algo que parecía una buena idea falla y es necesario volver a algo que se decidió abandonar en primer lugar. Una pequeña ampliación en la evaluación generalmente permitirá menos prueba y error, y en definitiva una mejor decisión en el largo plazo.

1.6.1 Evaluación de software libre.

El realizar evaluaciones al software libre no solo es una necesidad para proveer a una institución de software confiable y económicamente eficiente, sino que en muchos casos existe software libre que compite en igualdad o superioridad de condiciones con software propietario, e inclusive de no ser así el caso, la existencia de software libre ha generado competencia y exigencia para que las soluciones de pago tengan precios más razonables y mejores características e incluso ha ayudado al poder de negociación de los posibles usuarios (Mangalam, 2010).

1.6.2 Particularidades de la evaluación de software libre.

Existen varias razones principales hacen a una evaluación de software libre distinta a la tradicional de software propietario, pero antes de ser un limitante es generalmente ventajosa ya que está disponible mayor información del proyecto.

Para mayor claridad se ha realizado una comparación de particularidades en las características entre proyectos de software libre y propietario:

Tabla No.2: Comparación entre software libre y propietario

Características	Software libre	Software propietario
Documentación de desarrollo	El desarrollo de proyectos Software Libre de mediano o gran tamaño generalmente publica mayor cantidad de documentación acerca de su desarrollo y procesos	Los detalles del desarrollo del proyectos se mantienen ocultos por razones comerciales y de propiedad intelectual.
Promoción	Es escasa y proporcionalmente mucho menor en relación a los proyectos de software propietario. Se basan principalmente en promoción estilo "boca a boca" que los usuarios actuales hacen a potenciales candidatos, buscadores de internet y medios digitales como redes sociales con ningún o escaso presupuesto.	Estos proyectos disponen de presupuesto, equipos de ventas y marketing que preparan material de ventas que realizan campañas promocionales, lanzamientos, promociones, etc. que buscan resaltar las ventajas del producto.
Demos	Se puede por lo general probar una versión completa en línea o descargar y probar una versión completa sin limitaciones.	Generalmente con tiempo y funcionalidades limitadas, pueden requerir de un registro previo para su descarga.
Especificaciones	Se puede hacer una evaluación más profunda de sus características que también implica un esfuerzo mayor en investigar las mismas.	Generalmente dispone de personal y literatura de ventas que hacen su evaluación más sencilla y que ayudan bastante para su evaluación.

Características	Software libre	Software propietario
Enfoque	El mayor esfuerzo de los proyectos de software libre van hacia el desarrollo y la documentación, foros, fortalecimiento de la comunidad, etc.	Las empresas comerciales de software propietario manejan imagen de marca, publicidad, personal de ventas, incluso agencias de publicidad, etc. por lo que la percepción de calidad o adecuación puede verse afectada por estas particularidades (Mangalam, 2010)
Comunidad	En este tipo de proyectos la comunidad de usuarios y desarrolladores es una sola, y personas que inicialmente ingresaron como usuarios pueden involucrarse y volverse parte del equipo de desarrollo.	Existe una empresa de desarrollo que es separada de la comunidad de clientes usuarios, es una relación comercial, pero en algunos casos se pueden desarrollar ecosistemas de ampliaciones o extensiones de terceros en software que posea una gran popularidad.
Soporte	Existen una o varias empresas que ofrecen soporte comercial pagado para los proyectos que han obtenido mayor cantidad de usuarios activos, en los proyectos de menor madurez o popularidad es más difícil de conseguir.	Generalmente la empresa creadora ofrece varios niveles de soporte pagado o contratos anuales.

Fuente: elaborado por el autor

(Wheeler, 2011) entre otras particularidades importantes es que la naturaleza de los proyectos de Software Libre permita a todos la oportunidad de participar en su desarrollo, y existen muchos casos que usuarios de software libre se han involucrado y se han vuelto desarrolladores.

Igualmente hace énfasis en que el software libre tanto como el propietario forman parte de lo que se llama software comercial, pues clasificar este como no comercial es erróneo, ya que sus licencias y enfoque permiten su uso en actividades comerciales, y varios creadores de Software Libre ofrecen comercialmente opciones de asesoría y soporte pagados, por lo que la acepción correcta para este tipo de software es justamente el término “libre” o de “código abierto”.

CAPÍTULO 2 METODOLOGÍAS DE EVALUACIÓN DE SOFTWARE LIBRE

2.1 Inicios de las metodologías de evaluación

A pesar que los inicios claros del software libre se remontan hasta los años 70, no es hasta este siglo que se ha dado un desarrollo más completo de propuestas para métodos de evaluación de software libre que tomen en cuenta las características especiales de este tipo de software y también ante el crecimiento de las alternativas de este tipo de software en calidad y cantidad.

Al ser el desarrollo de proyectos de software libre una actividad en constante evolución, igualmente los métodos para su evaluación han ido apareciendo, modificándose y creando nuevos modelos que toman partes de uno o varios anteriores a este. A continuación se revisan algunas de las más recientes e importantes propuestas de metodología.

2.1.1 IRCA.

IRCA son las iniciales de Identify, Read (reviews), Compare and Analyze (Identificar, Leer evaluaciones, Comparar y Analizar) que tanto (Wheeler, 2011) como (Mangalan, 2010) mencionan en sus textos de métodos de evaluación de Software Libre.

2.1.1.1 Wheeler.

(Wheeler,2011) detalla sobre este proceso de evaluación propuesto lo siguiente:

- Identificar candidatos adecuados es un paso fundamental, investigando las opciones disponibles en especial aquellas que se consideran como GRAM Generally Recognized as Mature (Generalmente Reconocido como Maduro) y a través de sitios y organizaciones que cataloguen y evalúen software libre o que detallen software libre equivalente, análogo o sustitutos de programas de software propietario.

- Leer evaluaciones existentes sobre las alternativas disponibles, analizar la popularidad y nivel de uso de determinado paquete en organizaciones con requerimientos similares a la nuestra, pues productos con una buena participación del mercado son más fáciles de instalar y operar, además de hacer más viable su permanencia y desarrollo en el tiempo.
- Comparar las características principales de las aplicaciones en relación a nuestros requerimientos y también poder reducir la selección hacia un conjunto más reducido de candidatos. Para este paso los sitios web de los proyectos de software a comparar son una fuente importante de información para este paso para finalmente poder evaluar todos los proyectos con su respectivo software en relación a los atributos que se consideren importantes.
- Finalmente evaluar la funcionalidad es el proceso en el que se determina si el software puede hacer lo que requerimos que haga, además de que forma se integra y es compatible con los componentes actuales existentes.

2.1.1.2 Mangalan.

(Mangalan, 2010) destaca que la decisión de uso de Software Libre no solo se puede basar en el ahorro de recursos económicos sino que es una opción válida siempre que se busque proveer servicios TI eficientes y confiables. Su enfoque se basa en lo desarrollado por Wheeler con algunas adiciones:

Pone de manifiesto que es importante tanto para uso futuro propio como de otras personas el documentar el proceso para que pueda servir como material de referencia futura.

- Identificar. Resalta la importancia de usar herramientas como Wikipedia, búsquedas en Google y el investigar las comunidades online que se generan a partir de los proyectos de Software Libre.

- Leer evaluaciones. Una de las formas menos informativas es comparar dos paquetes a través de sus listas de características, que es una de las formas típicas utilizadas en técnicas de ventas para software de pago. Igualmente es común que en vez de buscar una solución grande de un solo proveedor, como en el software propietario, se busque el denominado enfoque de Unix con herramientas pequeñas que se especializan en hacer pocas cosas bien y trabajan adecuadamente en conjunto.
- Analizar. En lo posible instalar el software para un análisis a fondo, y de preferencia que exista la opción de disponer de contenido preinstalado para pruebas y evaluación. Muchas veces la instalación no es complicada, no así la configuración correcta de todos los parámetros, y dependerá mucho de la documentación disponible para el efecto. En general se debe tomar en cuenta que tanto la instalación como configuración se hace una sola vez.

Tabla No.3: Métricas cualitativas.

Curva de aprendizaje e interfaz de usuario
Facilidad de uso
Flujo de trabajo y lógica del programa
Documentación y ayuda
Uso de tecnología Web2/Javascript vs Web1/html

Fuente: Recuperado de Mangalan, 2010.

Tabla No.4: Métricas cuantitativas.

Raw Links en Google Trends
Tráfico en los foros del programa
Madurez
Escalabilidad
Complejidad de las bases de datos (Schemaball)
Soporte de plataforma

Fuente: Recuperado de Mangalan, 2010.

2.1.2 QSOS.

Qualification and Selection of Open Source Software, (Calificación y Selección de Software de Código Abierto) es una metodología iterativa para evaluación de Software Libre cuya esencia es ofrecer una metodología igualmente libre para el efecto .

Dispone de varias herramientas para generar y manipular documentos de evaluación QSOS en formato XML y de una comunidad para el desarrollo de material, evaluación y herramientas. Esta metodología es parte de un proyecto más grande llamado Drakkr (drakkr.github.io) que es una iniciativa de Gobernanza Abierta para ayudar a las organizaciones en la integración y manejo de componentes de software libre.

En su manifiesto (Atos, 2004) estipula que la necesidad de un método libre posibilite la reutilización de evaluaciones para cualquier persona, y mejorar la calidad y objetividad de la documentación generada que puede mejorarse a través de la transparencia y la revisión con iguales.

Este método consta de 4 pasos principales:

1. Definir y actualizar los criterios de evaluación, tanto en madurez como en características.
2. Evaluación del software en su cobertura funcional y madurez del proyecto
3. Calificación de criterios de acuerdo a un peso contextual
4. Comparación y selección del software basado en los datos de los pasos anteriores.

2.1.3 Qualipso OMM.

Es una metodología similar al modelo de integración de madurez de capacidades CMMI (integración de modelos de madurez de capacidades) para Software Libre que permita generar

confianza en el desarrollo de procesos en empresas usando o produciendo software libre. Su enfoque de uso va tanto hacia comunidades FLOSS, integradores, instructores y usuarios finales de sistemas de software libre (Qualipso, 2010).

Esta metodología está compuesta de tres niveles incrementales sobre el anterior que consideran elementos de confianza en el nivel más bajo, a continuación los elementos del nivel intermedio y finalmente los del nivel avanzado. Estos elementos y sus niveles son:

Tabla No.5: Nivel básico.

Documentación de producto
Uso de estándares establecidos y ampliamente reconocidos
Calidad del plan de testeo
Licencias
Ambiente técnico
Número de compromisos y reportes de bugs.
Mantenibilidad y estabilidad
Gestión de configuración
Planeación de proyecto 1
Gestión de requerimientos
Disponibilidad y uso de un hoja de ruta del producto 1

Fuente: recuperado de Qualipso, 2010.

Tabla No.6: Nivel intermedio.

Disponibilidad y uso de un hoja de ruta del producto 2
Relaciones entre Stakeholders
Planeación de proyecto parte 2
Monitoreo y control de proyecto
Testeo parte 1
Diseño parte 1
Aseguramiento de calidad de procesos y productos

Fuente: recuperado de Qualipso, 2010.

Tabla No.7: Nivel avanzado.

Integración de producto
Gestión de riesgos
Testeo parte 2
Diseño parte 2
Resultados de valoraciones de terceros
Reputación
Contribución de Producto Software Libre

Fuente: recuperado de Qualipso, 2010.

2.1.4 Van den Berg

En su tesis de maestría, Karin van den Berg plantea un método de evaluación de Software Libre que pueda ser usado ampliamente.

Esto es particularmente debido a que la mayoría de reportes de evaluación de software dan un sumario de sus resultados pero casi nada del método para llegar a estos, y debido a la rápida evolución del software en el tiempo los resultados de tales evaluaciones no son tan útiles como el método utilizado para obtenerlas .

La idea básica de su metodología es la utilización de ciertos criterios que forman la base del modelo basados, en principio, en las características únicas del software libre y que son relevantes para una evaluación de software.

Posteriormente se selecciona que información es necesario procesar para analizar los criterios seleccionados.

El conjunto de los criterios a usar en la evaluación provienen del análisis de varias metodologías

de evaluación FOSS existentes, seleccionando conceptos presentes en estas, muchas veces con distinta denominación.

En la práctica este modelo se puede resumir en dos pasos:

Inicialmente se realiza una selección de una lista de candidatos usando cuatro criterios:

- Funcionalidad, comparado con los requerimientos funcionales buscados.
- Comunidad, quienes se encuentran apoyando al proyecto.
- Historial de liberación de versiones del software
- Longevidad del proyecto desde su creación.

Estos criterios se toman en cuenta en esta primera etapa para seleccionar candidatos de forma rápida y así eliminar candidatos no viables generando una lista corta de los más adecuados para una segunda evaluación más a profundidad.

Posteriormente estos candidatos finales se los evalúa más en detalle tanto en los criterios anteriores como en otros nuevos:

- Licencia: tipo o tipos de licencias que utiliza
- Soporte: tanto de la comunidad como opciones de soporte pagado.
- Documentación: manuales, tutoriales y documentación de los desarrolladores.
- Seguridad: en respuesta a vulnerabilidades posibles
- Integración: a estándares, modularidad y colaboración con otros productos
- Objetivo y origen: como inició el proyecto y cuales son sus metas actuales.

(van den Berg, 2005)

2.1.5 Evaluación a través de Análisis de Proyectos

(Cruz, Wieland, & Ziegler, 2006) proponen un acercamiento racional que ayude al proceso de toma de decisiones para incorporar un producto de Software Libre en una empresa.

Su análisis se basa inicialmente determinar un escenario para la adopción de software FLOSS, basado en requisitos funcionales, técnicos, organizacionales, económicos y políticos.

Posteriormente se determina cuales de los requisitos buscados son los más importantes y se les asigna una valoración relativa proporcional.

Luego de esto se determina por cada tipo de requisitos varios aspectos a tomar en cuenta:

2.1.5.1 Requisitos funcionales.

Cobertura suficiente y un panorama claro de la evolución del software.

2.1.5.2 Requisitos técnicos.

Soporte para la plataforma escogida, confiabilidad y mantenibilidad.

2.1.5.3 Requisitos organizacionales.

Existencia de una comunidad, evolución activa del software, existencia del proyecto por varios años y proceso de desarrollo compatible.

2.1.5.4 Requerimientos legales

Que no existan efectos copyleft para añadidos o combinaciones, descargo de responsabilidad

por el código de terceros y no infringir ninguna patente.

2.1.5.5 Requisitos económicos

Sostenibilidad en el uso de Software Libre, protección de las inversiones para migrar, incremento de productividad por el uso del software, un esquema flexible de mantenibilidad de acuerdo a las necesidades empresariales, rápida disponibilidad de la aplicación, reducción de costos durante el uso del mismo y división de costos de desarrollo a través del uso del FLOSS.

2.1.5.6 Requerimientos políticos.

Es una de las partes más novedosas de este análisis y tiene conceptos como la posibilidad de influenciar futuros desarrollos del software de acuerdo a necesidades individuales, disminución de dependencias a software propietario, transparencia sobre la seguridad de la información, y reconocimiento público, efectos en publicidad y mercadeo cuando se usa y participa en un proyecto de Software Libre.

A partir de estos requisitos y subrequisitos se investiga y recolecta datos cualitativos y cuantitativos sobre los mismos y posteriormente generar un listado de nivel cumplimiento de cada requisito para el software candidato y una vez obtenido el mismo realizar una evaluación y toma de decisión contrastando esta información con el escenario descrito al inicio del proceso para determinar si los requerimientos esperados se cumplen, para que el documento desarrollado sirva como guía para ayudar a generar un análisis sistemático de los principales factores a tomar en consideración.

2.1.6 OpenBRR

El Business Readiness Rating for Open Source (Modelo de valoración de disponibilidad para

negocios de software libre) es un método creado en 2005 por Intel, O'Reilly Code Zoo, y la universidad Carnegie Mellon con el objetivo de construir un modelo estandarizado y facilitar la evaluación y adopción de software libre (Arjona, 2012).

En su más reciente guía (Openbrr.org, 2005) se detalla que la evaluación de software es una tarea crítica para la gestión de TI y que la falta de un procedimiento efectivo y confiable para la toma de decisiones sobre los mismos complica la adopción de software libre, pues las empresas difícilmente pueden aprender de otras experiencias similares en este aspecto.

Este modelo busca estandarizar varios tipos de datos para evaluar y los reúne en categorías. Este tipo de evaluación busca ser completo, simple, adaptable y consistente (CSAC).

Inicialmente se genera una primera revisión para filtrar y eliminar a los candidatos menos adecuados y así generar una lista corta con aquellos que son más viables y confiables.

Posteriormente se busca definir y estudiar las métricas cualitativas y cuantitativas que se pueden utilizar para el siguiente proceso de análisis más en profundidad, así como las formas de normalizar los valores obtenidos. Adicionalmente es importante organizar el proceso de valoración en categorías, para lo cual se han definido 12 ítems que deben ser valorados por su nivel de importancia y seleccionar al menos 7 y asignarles un peso específico en el proceso:

Tabla No.8: Ítems para tomar en cuenta en la evaluación OpenBRR

Funcionalidad
Usabilidad
Calidad
Seguridad
Rendimiento
Escalabilidad
Arquitectura
Soporte

Documentación
Adopción
Comunidad
Profesionalismo

Fuente: elaborado por el autor.

Para cada categoría se recopilan varias métricas, junto con sus peso específico de acuerdo a su importancia, que se relacionen directamente con esta y entre todas estas se obtiene una valoración de categoría.

A continuación se busca especificar el ajuste de la orientación funcional para la que el Software Libre está planeado, tanto de tipo de componente (por su utilidad) como por su tipo de uso (misión crítica, uso rutinario, desarrollo interno o experimentación) lo que nos dará una idea de la orientación funcional que se busca del software.

Finalmente una vez que se recopilen los datos de todas las métricas establecidas y se calcule sus pesos y valores se puede generar una sumatoria de todas las categorías que genere el índice OpenBRR a ser publicado.

2.1.7 Modelo de Madurez Smals OSS

Smals, una de las empresas de gestión de TI más importantes de Bélgica ha publicado con licencia Creative Commons lo que se puede considerar una evolución o versión mejorada de la metodología OpenBRR, con sus propias plantillas y un manual de uso.

En su guía de uso se sugiere que este modelo sea desarrollado por una persona de perfil técnico junto con una persona con un perfil más funcional, y también se incluye el criterio de licencia sin ponderación sino en el sentido de tomar en cuenta si es viable o no utilizar cierto tipo de licencias (Lannoy, 2009).

Este modelo añade cambios en las categorías de evaluación como:

- Tomar en cuenta indicativos de potenciales problemas en caso de bajos valores como los errores o bugs solucionados en los últimos 6 meses.
- Añadir edad y estado de liberaciones como métricas de Comunidad
- Añade la categoría `Licencia` con nuevas métricas como protección contra derivaciones propietarias, limitaciones de las ediciones libres o comunitarias del software, nivel de permisividad y licencias múltiples.
- Cambia Usabilidad por Instalación
- Retira la categoría Funcionalidad (Arjona, 2012)

2.2 Análisis comparativo de las metodologías

Una vez realizada la investigación de modelos de evaluación se procedió a una comparación y evaluación de los mismos a través de la documentación disponible que nos permite distinguir características similares entre la mayoría de estos:

- Una lista de proyectos de Software Libre aptos para nuestros objetivos es elaborada a través de investigación de varias fuentes.
- Posteriormente todos estos proyectos pasan a un proceso de eliminación que genera una lista de candidatos.
- Esta lista pasa a un proceso de selección que analiza todos los candidatos en una serie de criterios fundamentales que generan una valoración cuantitativa para cada software.
- Se realiza una evaluación a profundidad de los dos mejores proyectos y finalmente se recomienda el que tenga mejor desempeño global.

Tabla No.9: Comparación de características de metodologías de evaluación

Metodología	Ventajas	Desventajas
IRCA Wheeler/ Mangalan	<p>Es un método que aborda correctamente las particularidades de la evaluación de software libre.</p> <p>Destaca además la búsqueda de fuentes de información sobre proyectos FLOSS y determina ciertas métricas cualitativas y cuantitativas.</p> <ul style="list-style-type: none"> - Identificar - Revisar evaluaciones - Comparar características - Analizar y evaluar 	<p>No posee una estructura metodológica completa con criterios específicos, es una aproximación a este tema bastante libre y general, que no ahonda en los detalles a tomar en cuenta.</p>
QSOS	<p>Es una metodología iterativa libre.</p> <p>Dispone de herramientas XML y una comunidad para el desarrollo de material.</p> <ul style="list-style-type: none"> - Definir criterios - Evaluar - Calificar - Comparar y seleccionar 	<p>Desde 2013 no ha existido actividad notoria en la comunidad y la fortaleza de este método es el apoyo comunitario.</p> <p>En el caso de este trabajo la evaluación se tiene que hacer individualmente y no existen evaluaciones recientes sobre software de gestión bibliotecaria.</p>
Qualipso OMM	<p>Similar a CMMI.</p> <p>Se compone de tres niveles incrementales con distintos criterios de cumplimiento.</p> <ul style="list-style-type: none"> - Nivel básico - Nivel medio - Nivel avanzado 	<p>No es posible encontrar documentación pormenorizada de la metodología.</p>

Metodología	Ventajas	Desventajas
Van den Berg	<p>Es un modelo propuesto y documentado detalladamente en una tesis de maestría de la Universidad de Tillburg (Holanda) que toma los elementos comunes de varias metodologías de evaluación y desarrolla un modelo nuevo que se puede adaptar a cada caso de evaluación manteniendo un esquema de criterios, método y valoraciones bastante completo.</p> <ul style="list-style-type: none"> - Descripción de criterios - Proceso de búsqueda de proyectos - Ronda previa de eliminación - Selección de candidatos - Evaluación a profundidad de candidatos finales 	No ha evolucionado posteriormente.
Análisis de proyectos	<p>Es un enfoque con mayor énfasis como un proceso de toma de decisiones de acuerdo a un escenario que determina y valora:</p> <ul style="list-style-type: none"> - Requisitos funcionales - Requisitos técnicos - Requisitos organizacionales - Requisitos económicos - Requisitos políticos. 	Se centra en un proceso más corporativo en el que existe un comité interdepartamental que maneje el proceso de determinar el escenario, los requisitos y valoración de los mismos.
Open BRR	<p>Desarrollado por Intel, O'Reilly y Carnegie Mellon como un modelo estandarizado que facilite la evaluación y adopción de software libre.</p> <ul style="list-style-type: none"> - Revisión y filtrado de candidatos - Definir y estudiar métricas - Recopilar datos y generar el índice Open-BRR 	Es un método muy completo, pero el proyecto desde 2005 se ha detenido totalmente y no se puede obtener documentación al detalle del mismo.

Metodología	Ventajas	Desventajas
Smals OSS	Evolución de OpenBRR con algunas actualizaciones en criterios nuevos a tomar en cuenta.	La documentación existente se limita a una plantilla de hoja de cálculo para evaluación y un manual del modelo de 4 páginas que no permite un detalle mayor del modelo en sí.
CONCLUSIÓN	A lo largo de los distintos modelos y con nombres iguales o distintos se detallan una serie de criterios de eliminación, selección y evaluación, así como el peso específico que se puede asignar a cada uno de estos.	

Fuente: elaborado por el autor.

2.3 Elección del modelo a utilizar para la evaluación

Luego de un análisis de las necesidades particulares de este trabajo de titulación se ha considerado que la metodología propuesta por Karin van den Berg es la más idónea principalmente por las siguientes razones:

- Produce una integración más armónica, detallada y actualizada.
- Su tesis documenta detalladamente la metodología, criterios y motivaciones de la estructuración del sistema de evaluación.
- El modelo propuesto toma elementos de otras metodologías y las combina de una forma razonablemente adecuada.
- Se permite además realizar ajustes a criterios, pesos específicos en valoración y otros aspectos para adaptarse mejor a las particularidades de este caso específico.
- Se puede realizar una evaluación detallada y documentada que vaya más allá de llenar una plantilla de hoja de cálculo que arroje los resultados.
- Es un método perfectible y que puede ser un punto de partida para crear variaciones incrementales, al igual que en los proyectos de Software Libre, y de esta manera se puede subsanar de cierta forma el que no haya tenido evoluciones posteriores.

CAPÍTULO 3
CASO DE ESTUDIO: SOFTWARE LIBRE PARA BIBLIOTECAS
UNIVERSITARIAS EN ECUADOR

La presente evaluación es un análisis de las alternativas de software libre para gestión integral de bibliotecas (ILS, Integrated Library Systems en inglés), que al momento actual de este estudio (Octubre de 2016) cuentan con alternativas viables para gestionar bibliotecas de nivel universitario competitivamente en relación a los productos de código propietario.

El tamaño, infraestructura, personal y organización de una biblioteca universitaria en el país requiere de una adecuada selección que permita disponer de una efectiva solución de software a largo plazo, razón por la cual se realiza la presente evaluación.



Figura 2 Gráfico explicativo metodología a utilizar en la evaluación

Fuente: elaborado por el autor.

3.1 Introducción

De acuerdo a la Ley Orgánica de Educación Superior (LOES), el Consejo de Evaluación, Acreditación y Aseguramiento de la Calidad de la Educación Superior es el ente rector encargado de la evaluación y categorización universitaria en el país ha determinado en (CEAACES, 2015) dentro de los criterios de evaluación de recursos e infraestructura el subcriterio Biblioteca, y su primer indicador cualitativo es Gestión de la biblioteca, que consta de los siguientes elementos:

- Asignación de un presupuesto para el incremento del acervo bibliotecario.
- Políticas de funcionamiento y normativa de uso de la biblioteca.
- Responsable y personal de la biblioteca capacitado .
- Sistemas informáticos de gestión e información de la biblioteca.
- Procedimientos explícitos de articulación entre los servicios de la biblioteca y las necesidades académicas de la institución.
- Mediciones periódicas del uso de los recursos bibliográficos disponibles.
- Planificación y ejecución de la adquisición de recursos bibliográficos.

Este conjunto de criterio y subcriterios es el que pone de manifiesto la necesidad de que una biblioteca universitaria disponga de un software de gestión bibliotecaria que permita cumplir con los requerimientos detallados por el CEAACES.

Adicionalmente el decreto 1014 de 10 de abril de 2008 que establece como política para la Administración Pública la utilización de software libre, que se podría aplicar por extensión a todas las universidades públicas e incluso a aquellas que parcialmente reciben financiamiento estatal.

De una encuesta realizada para este trabajo de titulación y de la tabulación y análisis de sus resultados podemos determinar que la totalidad de bibliotecas universitarias del país dispone de algún tipo de software de base de datos para el manejo de su fondo bibliográfico, el número de títulos catalogados va de 3.500 a 109.000 y actualmente solo una universidad en el país, la Universidad Metropolitana del Ecuador, utiliza una solución de software libre (ver anexo B)

La mayoría de software de gestión de bibliotecas propietario exige una licencia individual para cada campus que se encuentre separado físicamente, por lo que una solución de software libre representa un ahorro de recursos importante, y tanto estos como el software libre requiere de un soporte de un departamento de TI, que en el caso de instituciones universitarias siempre existe y está disponible.

Existen experiencias de uso actual de software libre para gestión de bibliotecas en el sector público como la Biblioteca Nacional Eugenio Espejo, Cinemateca de la Casa de la Cultura Ecuatoriana, Sistema Nacional de Bibliotecas, Tribunal Contencioso Electoral, Consejo de la Judicatura, entre otros, que se vuelven referentes que generan experiencias y curvas de aprendizaje obtenidas de la implementación de este sistema en el ámbito nacional, que determina que existe la capacidad técnica y humana para su manejo adecuado.

Para definir el concepto de Sistema Integrado de Gestión Bibliotecaria (ILS), (Senso, 2011) manifiesta que:

Senso, 2011

Este tipo de programas surgen como un intento por conseguir que las unidades informativas (bibliotecas, centros de documentación o archivos) sean más eficaces, capaces de gestionar mejor sus recursos y de comunicarse de manera más práctica con los usuarios... integra, en un solo programa informático, a un conjunto de aplicaciones específicas -denominadas módulos- pensadas para facilitar el trabajo específico en cada una de las tareas típicas de estos centros.

Evidentemente, todas estas aplicaciones están interrelacionadas entre sí.

Para ello, emplean una misma base de datos de tipo relacional, con lo que pueden aumentar su eficacia, reducir la posibilidad de que existan informaciones repetidas y aumentar así la eficacia. La forma habitual de proceder de estas aplicaciones es siempre muy parecida: se emplea un módulo de administración, que se encarga de la configuración del sistema, de establecer los criterios de uso (tipo de préstamos, tipo de usuarios, clases de documentos...) y de gestión (compra de nuevos documentos, gestión de presupuesto, correspondencia, comunicación con proveedores...). Toda esa información se almacena en una misma base de datos, con el fin de que se pueda intercambiar información y aprovechar mejor el esfuerzo realizado. (p1)

Cabe indicar que el software de gestión bibliotecaria contempla aquel que permite la catalogación de ejemplares físicos de libros, revistas y similares con soporte en papel, y adicionalmente el de recursos digitales y que es distinto a un sistema de gestión de una biblioteca virtual o repositorio digital.

3.2 Selección

Para este proceso se realiza un listado de candidatos inicial y a continuación se realiza una eliminación por aspectos que nos permite reducir la lista a aquellos que cumplan con ciertos criterios base para continuar el proceso de selección.

3.2.1 Criterios de selección

Este modelo se basa en varios criterios que agrupan características que se buscan en un proyecto de software libre.

Tabla No.10: Criterios de evaluación metodología van den Berg

Criterio	Descripción
Comunidad	Es la principal fuente de recursos y fuerza que respalda un proyecto de Software Libre
Historial de versiones	Indicativo de las actividades de desarrollo y su progreso
Longevidad	Determina la estabilidad y sus posibilidades de supervivencia
Licencia	Si es pública o especializada, GNU GPL es la más común con copyleft, lo que asegura que el código y cualquier derivado mantenga el mismo tipo de licencia.
Soporte	Tanto comunitario como de pago, para responder preguntas como para soporte de fallos
Seguridad	Transparencia en vez de reserva, es una característica que debe considerarse muy seriamente
Funcionalidad	Liberaciones tempranas y muy seguidas pueden generar productos incompletos. La información acerca de las características no siempre es clara.
Integración	Arquitectura modular, uso de estándares y capacidad colaborativa con otros productos.
Objetivos y origen	Es importante que el objetivo del proyecto FLOSS encaje con la intención de uso y da una idea general de la seriedad y compromiso de los desarrolladores con el proyecto.

Fuente: (van den Berg, 2005)

3.2.2 Métricas cualitativas y cuantitativas

Tabla No.11: Métricas cualitativas y cuantitativas

Criterio	Selección	Evaluación
Comunidad	Número de post y usuarios	Número de posts, totales y por período, número de usuarios y desarrolladores, velocidad y calidad de las respuestas, contribuciones de la comunidad (código, documentación, etc)
Historial de liberaciones	Conteo de versiones y significancia	Versiones por período y significancia, mapa de ruta (roadmap)
Longevidad	Tiempo desde la primera versión	Edad, versión, ha pasado por ciclos de rediseño? El número de versión no es garantía
Licencia		Licencia pública (GNU) El tipo de licencia encaja con el uso planificado?
Soporte		Actividad de la comunidad y calidad, opciones pagadas del vendedor o de terceros
Seguridad		Recomendaciones de seguridad y rastreador de bugs (bugtracker) Tiempo de respuesta a huecos de seguridad

Funcionalidad	Chequeo con requerimientos mínimos mediante lista de características, búsqueda de desarrolladores y comunidades, demos de aplicaciones web	Chequeo de lista de características, preguntar a desarrolladores, demo en línea, testeo con instalación. Comparar con requerimientos funcionales. Chequeo de riqueza de características, funcionalidades extra que pueden ser útiles, estas dan una indicación de audiencia objetivo y el entusiasmo de los desarrolladores.
Integración		Chequear la estructura del código y su documentación para modularidad. Se están implementando estándares relevantes? Trabaja con productos relevantes? Qué nivel de flexibilidad tiene con las versiones de software requeridas?
Objetivos y origen		Chequear documentación para su historia y mapa de ruta.

Fuente: elaborado por el autor

3.2.3 Lista de candidatos

Para la elaboración de esta lista se realizó una búsqueda en varias fuentes sobre las opciones de software libre disponibles para gestión integral de bibliotecas, en base a:

- Publicaciones y artículos especializados
- Recomendaciones de expertos en manejo de bibliotecas (Bibliotecología)
- Evaluaciones de software
- Tendencias en internet

Tabla No.12: Lista de candidatos

Nombre	Sitio web del proyecto
CDS/ISIS Winisis	http://portal.unesco.org/ci/en/ev.php-URL_ID=5330&URL_DO=DO_TOPIC&URL_SECTION=201.html#doc
ABCD	http://abcdonline.com.ar/ http://abcdwiki.net/wiki/es/index.php?title=P%C3%A1gina_principal http://abcdwiki.net/ABCofABCDv13t/index.html
Aguapey	http://www.bnm.me.gov.ar/redes_federales/aguapey/
Biblioteq	http://sourceforge.net/projects/biblioteq
Catalis	http://inmabb.criba.edu.ar/catalis/catalis.php?p=main
Clabel	http://clabel.sourceforge.net/
Elibrary	http://sourceforge.net/projects/ebooklibrary/
Espabiblio	http://sourceforge.net/projects/espabiblio/
Evergreen	http://evergreen-ils.org/
GreenStone	http://www.greenstone.org/
Kobli	http://kobli.bage.es/
Koha	http://koha-community.org/
Koha UNPL	http://www.koha.unlp.edu.ar/
Meran	http://www.cespi.unlp.edu.ar/meran
NewGenLib	http://sourceforge.net/projects/newgenlib/
Open Flora	http://www.openflora.com/
Open Marco Polo	http://marcopolo.uner.edu.ar/index.htm
OpenBiblio	http://sourceforge.net/projects/obiblio/
PMB	http://www.pmbservices.fr/
RefBase	http://sourceforge.net/projects/refbase/
SOPAC	https://www.drupal.org/project/sopac
WebLis	http://portal.unesco.org/ci/en/ev.php-URL_ID=16841&URL_DO=DO_TOPIC&URL_SECTION=201.html

Fuente: elaborado por el autor

3.2.4 Eliminación por aspectos

En este paso del proceso es necesario eliminar a los candidatos que no cumplan con alguno o algunos de los criterios mínimos establecidos. Entre otras características de eliminación se ha seleccionado las siguientes:

- Que su desarrollo o soporte se encuentre discontinuado (más de 2 años).
- Escasa o nula disponibilidad de versión en español.

- Que el código fuente no esté disponible.
- Que sea una versión de prueba (beta) no apta para entornos de producción.
- No cumplir con las funcionalidades completas de un sistema integral de gestión bibliotecaria, como módulos de trabajo que permitan cumplir los requerimientos que (CEAACES, 2015) detalla, tales como:
 - adquisiciones
 - catalogación
 - publicaciones seriadas
 - gestión de usuarios y circulación
 - reportes y estadísticas
 - administración
 - OPAC (Online Public Access Catalog / Catálogo en línea de acceso público).

Tabla No.13: Lista de eliminación/ paso a ronda final

Nombre	Versión	Ultima actualización	Ronda final	Observaciones
ABCD	1.5	2016-01-21	No	Basado en CDS ISIS, está en proceso de convertirse en open surce
Aguapey	1.3.3	2002	No	Basado en Isis. No es software libre y ya no se actualiza. Solución desarrollada para bibliotecas de la administración pública de Argentina.
Biblioteq	2016.02.17	2016.02.17	No	No existe versión en español. Es una base de datos de libros más que un sistema integral de gestión de bibliotecas
Catalis	2005.06.09	2005	No	La última actualización es de 2010. Es una herramienta para crear catálogos de bibliotecas más que un sistema de gestión integral. Descontinuado

Nombre	Versión	Ultima actualización	Ronda final	Observaciones
CDS/ISIS Winisis	1.5 build 3	2008	No	Se ha dejado de desarrollar. Solo disponible para windows, no está soportado por Windows 7 o superior ni en 64 bits.
Clabel	0.5.0	2005	No	Descontinuado. No se actualiza desde 2005. Es una herramienta que permite acceder a bases de datos ISIS desde el ambiente web.
Elibrary	2.5.13	2009-10-09	No	Es un organizador de documentos digitales personales con ciertas características para clasificar libros físicos. Descontinuado.
Espabiblio	3.3 basado en Openbiblio 7.1	2015-03-24	No	Basado en OpenBiblio 7.1 beta
Evergreen	2.10.1	22/03/2016	Si	
GreenStone	3.07	9 Sep 2015	No	Es un software para gestión de bibliotecas o repositorios virtuales y no físicos
Kobli	1.12.4	05/12/13	Si	Basado en Koha, adaptado para las bibliotecas de la administración pública de España, por lo que la liberación de versiones es más espaciada que el proyecto madre.
Koha	3.22	November 26, 2015	Si	
Koha UNPL	2.2.0	2010	No	Basado en Koha. A partir de 2012 el proyecto Koha UNPL cerró y se dedicó a desarrollar un proyecto propio: Meran
Meran	0.10.4	09/02/15	Si	
NewGenLib	3.1.2	2015-09-02	Si	
Open Flora	1.1.0	2011	No	Software y documentación disponible solo en francés.
Open Marco Polo	1.6	2008	No	Última actualización en 2008. Desarrollado por la Universidad Nacional de Entre Ríos, Argentina.
OpenBiblio	0.7.2 beta	2014-08-13	No	Se mantiene en versión beta y está
PMB	4.2	2015-12-11	Si	
RefBase	0.9.6	2014-02-28	No	Es una plataforma para importar y exportar referencias web y bibliográficas más que un sistema de gestión bibliotecaria.
SOPAC	6.x-2.0	December 24, 2010	No	Descontinuado, es un plugin del CMS Drupal para integrar un ILS dentro de los contenidos.
WebLis	70826	2007	No	Basado en CDS ISIS. Descontinuado desde 2007

Fuente: elaborado por el autor

A partir de la lista de eliminación, 6 de los 22 candidatos cumplen los requisitos establecidos para acceder a la siguiente ronda de valoración. La lista de candidatos para el proceso de selección es el siguiente:

Tabla No.14: Lista de candidatos seleccionados

Nombre	Versión
Evergreen	2.10.1
Kobli	1.12.4
Koha	3.22
Meran	0.10.4
NewGenLib	3.1.2
PMB	4.2

Fuente: elaborado por el autor

3.2.5 Valoración de los candidatos seleccionados

3.2.5.6.1 Peso específico de criterios de selección

Una vez seleccionados los candidatos, los cuatro criterios de selección se les asignará un peso determinado. La distribución porcentual para cada concepto de selección puede adaptarse a cada evaluación de acuerdo a sus particularidades, y en este caso se ha adoptado la idea de (van den Berg, 2005) , de tal forma que sobre 100 puntos asignar 10 a longevidad y los restantes 90 repartirlos en proporción decreciente entre los otros 3 de acuerdo a su importancia, de la siguiente forma:

Tabla No.15: Peso porcentual de criterios de selección

Criterio	Valor
Funcionalidad	35
Comunidad	30
Liberación de versiones	25
Longevidad	10
Total	100

Fuente: elaborado por el autor

3.2.5.1 Valoración de Funcionalidad

Para calificar este criterio se hizo una revisión de las funcionalidades de los candidatos a través de su listas de características y mediante el uso de entornos de demostración en línea disponibles realizando tareas de catalogación y administrativas típicas en un flujo de trabajo de una biblioteca universitaria. Para determinar el nivel y profundidad de las características existentes en cada software se utilizó los criterios de las tablas de características de (Café et al., 2001).

Adicionalmente a la metodología de van den Berg se ha desglosado la funcionalidad en detalles de: arquitectura, licencia, proyecto, tecnologías, estándares y módulos disponibles.

3.2.5.1.1 Arquitectura

Detalla que sistema de computación distribuida soporta el programa, y la o las distintas versiones disponibles en qué plataformas y sobre qué base de datos. Esta información es importante para poder determinar si la infraestructura actual o planificada de la institución converge con los requerimientos del software o requiere de cambios.

3.2.5.1.2 Licencia

Es el tipo y versión de documento que detalla el uso de los derechos de propiedad intelectual del software a utilizar.

3.2.5.1.3 Tecnología

Hace una descripción resumida de el o los lenguajes de programación utilizados en el proyecto y el motor de indexación usado.

3.2.5.1.4 Estándares

Enumera los protocolos de comunicación que el proyecto utiliza para comunicarse y obtener

información hacia fuentes externas.

3.2.5.1.5 Módulos

Lista las partes principales en las que está compuesto el software de gestión de bibliotecas.

Tabla No.16: Valoración de Funcionalidad

Proyecto	Puntos		Observaciones
Evergreen	8	Arquitectura	Aplicación cliente servidor sobre base de datos PostgreSQL con versión para servidor (Linux) y para clientes de administración Java (Windows, Mac, Linux). Cliente web en futuras versiones (actualmente preview). El número de versión del cliente como servidor deben ser iguales.
		Licencia	GNU GPL Ver. 2.
		Proyecto	Iniciado por el sistema de Bibliotecas Públicas de Georgia en 2006 y que se usa especialmente en EEUU y Canadá.
		Tecnologías	Basado en el framework Open SRF (Open Scalable Request) Framework. Tecnologías: Perl, C, JavaScript, XML, XPath, XSLT, XMPP, OpenSRF, Apache, mod_perl, and PostgreSQL. Buen nivel de escalabilidad
		Estándares	CODABAR, EDI, LDAP, MARC21, MFHD, OpenSearch, OpenURL, RDA, RSS, schema.org, SIP2, unAPI, Unicode, WCAG, Z39.50
		Módulos	Adquisiciones, Catalogación, Seriales, Circulación, Reportes, Administración, OPAC (catálogo público)
Kobli	8	Arquitectura	Aplicación web sobre base de datos MySQL
		Licencia	GNU GPL Ver. 3
		Proyecto	Basado en Koha y adaptado por el Ministerio de Cultura de España a través de Grupo de Trabajo de las BAGEs (Bibliotecas de la Administración General del Estado) para su uso en las bibliotecas públicas de ese país.
		Tecnologías	Escrito en Perl, indexador Zebra
		Estándares	MARC 21, UNIMARC, z39.50, SRU/SW, SIP2, SIP/NCIP, XML, Dublin Core, RSS, MARCXML, EndNote
		Módulos	Adquisiciones, Catalogación, Circulación, Publicaciones periódicas, Informes, Autoridades, Administración, OPAC (catálogo público)

Proyecto	Puntos		Observaciones
Koha	10	Arquitectura	Aplicación web sobre base de datos MySQL
		Licencia	GNU GPL Ver. 3
		Proyecto	Creado en 1999 por Katipo Communications (Nueva Zelanda)
		Tecnologías	Escrito en Perl, indexador Zebra
		Estándares	MARC 21, UNIMARC, z39.50, SRU/SW, SIP2, SIP/NCIP, XML, Dublin Core, RSS, MARCXML, EndNote, LDAP
		Módulos	Adquisiciones, Catalogación, Circulación, Publicaciones periódicas, Informes, Autoridades, Administración, OPAC (catálogo público)
		<i>Este proyecto cumple completamente todos los requisitos funcionales analizados</i>	
Meran	6	Arquitectura	Aplicación web sobre base de datos MySQL
		Licencia	GNU GPL Ver. 3
		Proyecto	Creado por el Centro Superior para el Procesamiento de la Universidad de la Plata, Argentina a partir de 2009 que anteriormente desarrolló KOHA-UNPL.
		Tecnologías	Escrito en Perl, indexador Sphinx 2
		Estándares	XML, MARC21
		Módulos	Catálogo, OPAC, Reportes, Circulación, Sistema
		Carece de comunicación con protocolo Z39.50 y algunos no posee módulo <i>específico</i> de publicaciones periódicas.	
NewGenBib	6	Arquitectura	Aplicación web sobre Apache Tomcat con base de datos PostgreSQL y Solr Lucene
		Licencia	GNU GPL
		Proyecto	Desarrollado por Versus Solutions en Hydereabd, India, iniciado en 2005
		Tecnologías	Java SE
		Estándares	MARC-21, MARC-XML, z39.50, SRU/W, OAI-PMH, Z39.50
		Módulos	Adquisiciones, Procesos técnicos (catalogación), Publicaciones seriadas, Circulación, Administración, Reportes, Agenda de tareas, OPAC.
		Sus módulos no tienen todas las características presentes en otros proyectos.	
PMB	10	Arquitectura	Aplicación web sobre plataforma LAMP (Linux, Apache, MySQL, PHP)
		Licencia	CeCILL (licencia de software libre adaptada a la legislación francesa) compatible con GNU GPL
		Proyecto	Iniciado en 2002 por François Lemarchand, actualmente la empresa francesa PMB Services lidera el proyecto
		Tecnologías	Escrito en PHP
		Estándares	UNIMARC, XML, Z39.50, SRU/SRW, OAI, FRBR, RDA, RDF, LDAP
		Módulos	Circulación, Catálogo, Informes, SDI (Diseminación selectiva de información), Administración, Adquisición, Administración, OPAC
		<i>Este proyecto cumple completamente todos los requisitos funcionales analizados</i>	

Fuente: elaborado por el autor

3.2.5.2 Valoración de Comunidad

La o las comunidades de cada proyecto de software son las organizaciones esenciales para su funcionamiento y crecimiento, y el estado de las mismas, por lo general es un fiel reflejo de la salud del proyecto en sí. Estos son los sitios donde se puede buscar ayuda, documentación, reportar bugs, compartir experiencias, etc.

Como valores cuantitativos que reflejan el desempeño comunitario de cada candidato se han establecido los tópicos y/o posts disponibles históricamente a lo largo de la vida del proyecto, ya que esto provee una valoración objetiva a lo largo del tiempo. La valoración se ha asignando a quien posee el mayor valor la más alta calificación y valores inferiores en proporción relativa. Adicionalmente, una comunidad que maneje español como idioma principal se ha añadido dos puntos adicionales en relación a aquellas que usan otro idioma por la facilidad que esto representa para este caso de estudio.

Tabla No.17: Valoración de Comunidad

Proyecto	Puntos	Tópicos	Posts	Observaciones
Evergreen	8		24.148	Gestionado en Markmail.org desde marzo 2006 en idioma inglés
Kobli	8		16.756	Gestionado en listserv.rediris.es en idioma español 412 posts en kobli-sysadmin@listserv.rediris.es desde mayo 2013 16.344 posts en kobli@listserv.rediris.es desde mayo 2011
Koha	10		275.411	Gestionado en Markmail.org desde 2004 en idioma inglés
Meran	3	8	75	Gestionado en Github desde 2013 en idioma español. Está en proceso la creación de una plataforma Moodle para gestión de la comunidad de usuarios.
NewGenLib	5		1371	Gestionado en Naddle Free Forum desde 2009 en idioma inglés
PMB	8		27.877	Gestionado con MailMan desde 2006 en idioma francés
Cuantificación de tópicos y posts a 1 de abril de 2016				

Fuente: elaborado por el autor

3.2.5.3 Valoración de Liberación de Versiones

El historial de liberación de versiones (releases) es un indicador claro de la continuidad en el desarrollo y corrección de errores de los proyectos analizados. Al tener la gran mayoría de candidatos liberaciones recientes se ha calificado la mitad de la valoración con este concepto y la mitad restante de acuerdo a una proporción directa al número de versiones liberadas desde el inicio del año 2014 (28 meses al momento de realizada la evaluación)

Tabla No.18: Valoración Versiones

Proyecto	Puntos	Última liberación	2014 a 2016	Observaciones
Evergreen	8	22/03/2016	56	de la versión 2.5.2 a 2.10.1
Kobli	0	05/12/2013	0	no existen nuevas versiones desde 2013
Koha	10	21/03/2016	84	de la versión 3.14.02 a 3.22.05
Meran	3,5	9/02/2015	9	de la versión 0.9.6 a 0.10.4
NewGenLib	5	10/3/2016	5	de la versión 3.0.4 a 3.1.3
PMB	6	1/04/2016	18	de la versión 4.0.10 a 4.2.7
Cuantificación de versiones desde enero de 2014 a abril de 2016				

Fuente: elaborado por el autor

3.2.5.4 Valoración de Longevidad del Proyecto

Esta característica busca dar una significancia mayor al tiempo que el proyecto tiene en marcha desde su primera liberación ya que se puede inferir que el tiempo ayuda a generar un proyecto más maduro, completo y con un número de fallas decreciente.

Tabla No.19: Valoración de Longevidad

Proyecto	Puntos	Primera liberación	Observaciones
Evergreen	6	29/11/2006	
Kobli	3	15/6/2011	
Koha	10	3/1/2000	Es el software más longevo de la lista, nació como una iniciativa para crear un programa de gestión nuevo ya que el que utilizaba tenía un bug y2k.
Meran	4	1/3/2010	

Proyecto	Puntos	Primera liberación	Observaciones
NewGenLib	4	23/06/2010	
PMB	8	1/12/2003	

Fuente: elaborado por el autor

3.2.6 Resultados de la selección

Una vez hecha la valoración de los candidatos se procede a reunir los valores de los cuatro criterios de evaluación y se les dá el peso específico determinado en la tabla 12 para obtener las calificaciones finales.

Tabla No.20: Resumen de resultados

Proyecto	Funcionalidad	Comunidad	Liberación versiones	Longevidad	Total con peso porcentual
Peso porcentual	35%	30%	25%	10%	sobre 10
Evergreen	7	8	8	6	7,45
Kobli	8	8	0	3	5,50
Koha	10	10	10	10	10
Meran	6	3	3,5	4	4,275
NewGenLib	6	5	5	4	5,25
PMB	10	8	6	8	8,20

Fuente: elaborado por el autor

De la tabla anterior podemos determinar el siguiente orden de calificaciones en orden descendente:

Tabla No.21: Posiciones en valoración final

Proyecto	Valoración final
Koha	10
PMB	8,20
Evergreen	7,45
Kobli	5,5
NewGenLib	5,25
Meran	4,275

Fuente: elaborado por el autor

En los resultados de la tabla No. 21 podemos observar que los 3 proyectos con menor calificación final tienen una diferencia considerable con los 3 primeros ya que sus funcionalidades y comunidad especialmente se encuentran mucho menos consolidadas que los 3 primeros candidatos. El proyecto Kobli si bien es una derivación (fork) de Koha de hace varios años se le concibió para servir específicamente en la administración de las bibliotecas generales del Estado en España y no ha sido evolucionado posteriormente.

Por otro lado el proyecto Meran es la primera experiencia de desarrollo de software libre para gestión de bibliotecas universitarias en Latinoamérica y su equipo de desarrollo anteriormente creó la versión derivada Koha UNPL por lo que es a mediano plazo un prospecto a tomar en cuenta para reevaluarlo.

Los tres primeros proyectos de la lista se pueden considerar los más maduros, y justamente también muestran características que los diferencian como su origen: Nueva Zelanda (Koha), Francia (PMB) y Estados Unidos (Evergreen), e igualmente en los lenguajes principales de programación utilizados: Perl (Koha), PHP (PMB) y Java (Evergreen).

De lo realizado en esta etapa de selección se determina entonces que los candidatos finales para la evaluación de sistema de gestión integral de bibliotecas universitarias son los proyectos Koha y PMB.

CAPÍTULO 4
CASO DE ESTUDIO PARTE 2: EVALUACIÓN KOHA Y PMB

Una vez culminada la etapa de selección en el capítulo anterior se procede a la evaluación de los dos candidatos finales: Koha y PMB.

4.1 Introducción

4.1.1 KOHA

Koha es un término de la etnia Maorí que significa regalo o donación, desarrollado inicialmente por Katipo Communications para la Corporación de Bibliotecas de Horowhenua (HLT, Horowhenua Library Trust) en Nueva Zelanda a inicios del año 2000 e inmediatamente se decidió liberarlo bajo la licencia GPL.



Figura 3 Logotipo de Koha

Fuente: https://wiki.koha-community.org/wiki/The_Koha_Logo:_Origins_and_Explanation

Como lo reseña (Eyler, 2003), a partir de entonces el proyecto ganó relevancia en las bibliotecas de Nueva Zelanda y en ese mismo año empezó a evaluarse e implementarse en Columbia Británica (Canadá) y Estados Unidos, en especial por los recortes presupuestarios que las bibliotecas se hallaban experimentando, y estos primeros usuarios iniciaron la colaboración y enriquecimiento del proyecto que desde entonces ha ampliado su utilización a todos los continentes en bibliotecas de todo tipo y tamaño.

Desde 2001, con la colaboración de una comunidad creciente se añadieron más funcionalidades al producto y sobre todo soporte a varios lenguajes, así como soporte de

estándares bibliotecológicos MARC, Z39.50, entre otros. Posteriormente varias empresas de algunas partes del mundo han colaborado al proyecto y ofrecen soporte pagado como LibLime, Bywater, Biblibre, entre otros.

En 2011 el Ministerio de Educación de España desarrolló un proyecto basado en Koha: Koblí para su uso en las bibliotecas estatales españolas.

4.1.2 PMB

PMB, originalmente PHPMyBibli, es un proyecto de origen francés a partir del año 2002, creado por Francois Lemarchand, director de la biblioteca de Agneaux (Baja Normandía), y junto a Eric Robert desarrollan el proyecto hasta convertirlo en un sistema integrado de gestión de bibliotecas (ILS) que es presentado como versión 1.0 en una conferencia internacional en Rabat (Marruecos). Posteriormente en 2003 se crea la empresa PMB Services para el desarrollo del proyecto a largo plazo.



Figura 4 Logotipo de PMB

Tomado de http://www.sigb.net/index.php?lvi=cmspage&pageid=2&id_logiciel=18

Sus principales características incluyen el soporte de catalogación para todo tipo de documentos con archivos adjuntos, manejo de tesauros, disseminación selectiva de la información, servidor

de archivos OAI (Open Archives Initiative), interfaz RFID, búsqueda de texto completo en adjuntos, servicio de OCR y API para el CMS Drupal, entre otros.(Open Source Guide, 2016)

4.1.2.1

4.2 Comunidad

4.2.1 KOHA

Este proyecto dispone de una grande y diversa comunidad que dispone de varios recursos para la interacción entre sus integrantes:

Dispone de una canal IRC en inglés (<http://irc.koha-community.org/koha/>) desde agosto de 2004. Se realizó un muestreo de cantidad de mensajes de chat diarios que arrojó un promedio de 539 mensajes diarios con una mínima de 40 y máxima de 1209.

En Bugzilla (<https://bugs.koha-community.org/bugzilla3/>), una herramienta web de seguimiento de errores existen a la fecha 16.232 reportes de bugs, el primero fue llenado el 2002-06-15, y el último en 2016-04-08.

En Github y Gitweb dispone de repositorios de control de versiones, e igualmente en Open Hub (<https://www.openhub.net/p/koha>) detalla un sumario de características importantes de el proyecto dentro de las cuales vale mencionar:

Lenguajes usados: SQL 38%, Perl 38%, Javascript 12%, otros (9) 12%

Tiene 26,816 commits (cambios) realizados por 366 personas que representan un estimado de 694,214 líneas de código.

Maneja un Dashboard (Interfaz gráfica de usuario) para el control de versiones, agendamiento de la siguiente versión, QA, control de nuevas funcionalidades, versionado y bugs, con conexión a Bugzilla, Wiki, Jenkins y Sandboxes.

El wiki del proyecto tiene como última modificación el 20 de febrero de 2016 y ha sido visitado 667.888 veces. Existe un listado de bibliotecas usuarias de Koha en Sudamérica en <https://wiki.koha-community.org/wiki/KohaUsers/SouthAmerica>.

Adicionalmente cabe destacar que este proyecto dispone de un código de conducta para los integrantes de su comunidad, un manual del desarrollador con un nivel de detalle importante de nivel de colaboración, reglas, flujos de trabajo, etc.

Adicionalmente Koha dispone de comunidades en varios idiomas (francés, chino, italiano y español) y anualmente se desarrolla el KohaCon, una convención mundial de usuarios y desarrolladores de Koha que se realiza anualmente desde 2010 y que ha cubierto hasta el momento:

KohaCon16 – Thessaloniki, Grecia
KohaCon15 – Ibadan, Nigeria
KohaCon14 – Córdoba, Argentina
KohaCon13 – Reno, Nevada, USA
KohaCon12 – Edinburgo, Escocia
KohaCon11 – Thane, India
KohaCon10 – Wellington, Nueva Zelanda

4.2.2 PMB

Este proyecto cuenta con varias herramientas dentro de su comunidad, mayoritariamente en idioma francés:

Listas de correo especializadas, pmb-user para los usuarios en general del programa, pmb-trans para quienes desean traducir la interfaz de pmb a su idioma, y pmb-cdi para quienes utilizan el software en una escuela o centro educativo.

Forge donde se gestiona el proyecto PMB y otros anexos al mismo, dispone de actividad, miembros (582), planificación, peticiones (123), foros (sin uso), wiki y archivo.

Además existe un grupo de usuarios basado en Bélgica PMB-BUG (http://www.patch-works.be/_pmb/) que maneja noticias, actividades y documentación.

Adicionalmente existe el sitio de Anne-Marie Cubat (<http://amcubat.be/docpmb/>) que dispone de muchos recursos, tutoriales en gran cantidad de aspectos del manejo de una biblioteca mediante PMB.

AGESIC, organismo estatal que busca la implementación del gobierno electrónico en Uruguay (<http://www.agesic.gub.uy/>) recomienda la utilización de PMB como software para bibliotecas en la administración pública nacional.

4.2.3 Calificaciones

Tabla No.22: Resumen calificaciones comunidad

Criterio:	Comunidad	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	9	La comunidad de Koha tiene un tamaño y una distribución en el mundo más amplia, con mayores recursos y herramientas, mientras que PMB sus comunidades se concentran mayoritariamente en Francia y Bélgica.
PMB	6	

Fuente: elaborado por el autor

4.3 Liberación de versiones

4.3.1 Koha

Este proyecto se maneja con un ciclo de liberación de versiones basado en tiempo:

- Nuevas funcionalidades son liberadas cada 6 meses, liberaciones x.x.0
- Cada mes se libera versiones con arreglos de defectos (bugs) x.x.1 , x.x.2, etc

Las liberaciones funcionales han sido 36:

Versión 1.0 a 1.9

Versión 2.0 a 2.2

Versión 3.0 a 3.22 (actual)

4.3.2 PMB

Han existido 21 versiones funcionales liberadas desde la salida de la versión 1.0, con el siguiente detalle:

Versión 1.0 a 1.3

Versión 2.0 a 2.7

Versión 3.0 a 3.5

Versión 4.0 a 4.2 (actual)

4.3.3 Calificaciones

Tabla No.23: Resumen calificaciones liberación de versiones

Criterio:	Liberación de Versiones	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	10	Koha se le asignó 10 por ser el que más versiones ha liberado y se puntuó a PMB proporcionalmente.
PMB	6	

Fuente: elaborado por el autor

4.4 Longevidad

4.4.1 edad y version

4.4.1.1 Koha

En enero del año 2.000 se liberó la versión 1.0 y actualmente se encuentra en la versión 3,22

4.4.1.2 PMB

En octubre de 2003 se completa la versión inicial 1.0 hasta la versión 4.2 de la actualidad.

4.4.2 Calificaciones

Tabla No.24: Resumen calificaciones longevidad

Criterio:	Longevidad	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	10	En este criterio de evaluación no existen diferencias marcadas, y los años de vida tanto de Koha (16) y PMB (12) configuran proyectos maduros ya que por regla general un proyecto de más de 10 años se puede considerar maduro.
PMB	10	

Fuente: elaborado por el autor

4.4.3 Nuevas tecnologías

4.4.3.1 Koha

Koha está escrito en lenguaje Perl en su gran mayoría, con SQL como lenguaje para el manejo de las tablas en la base de datos (MySQL de preferencia)

4.4.3.2 PMB

Como su nombre lo indica, este proyecto utiliza lenguaje PHP y base de datos MySQL

4.4.3.3 Calificaciones

Tabla No.25: Resumen calificaciones nuevas tecnologías

Criterio:	Nuevas tecnologías	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	8	Ambos proyectos manejan MySQL como base de datos, y ambos se encuentran escritos en lenguajes bastante robustos y adecuados para el desarrollo de aplicaciones web (Perl para Koha y PHP en PMB).
PMB	8	

Fuente: elaborado por el autor

4.5 Licencias

4.5.1 Koha

Dispone de una licencia de software libre GNU GPL versión 3.

4.5.2 PMB

Si bien la licencia CeCILL, desarrollada por varias agencias francesas (CEA CNRS INRIA Logiciel Libre), es compatible con GNU pues se guía por los principios que regulan el Software Libre, su utilización denota un enfoque primeramente local ya que define a la ley francesa como el marco en el cual se refiere a la propiedad intelectual y la protección que esta ofrece a los autores y poseedores de los derechos económicos sobre el software.

4.5.3 Calificaciones

Tabla No.26: Resumen calificaciones licencias

Criterio:	Licencias	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	10	Prácticamente iguales, la licencia CeCILL es compatible con GNU.
PMB	10	

Fuente: elaborado por el autor

4.6 Soporte

4.6.1 Comunidad

4.6.1.1 Koha

Koha dispone de soporte sin costo en varias comunidades con un alto nivel de actividad, mediante las herramientas del proyecto principal y también en sus comunidades en francés, chino, italiano y español.

4.6.1.2 PMB

Este proyecto ofrece soporte comunitario a través de las listas de correo en PMB Services..

4.6.2 Soporte pagado

4.6.2.1 Koha

Koha oficialmente lista 57 empresas que ofertan soporte pagado, distribuidos en todos los continentes:

Africa:	4
Asia:	17
Europa:	23
Nueva Zelanda/ Australia:	6
América del Norte:	3
América del Sur:	5

Los principales proveedores de soporte dentro de la comunidad Koha son:

LibLime Koha (EEUU)
 ByWater Solutions (EEUU)
 Biblibre (Francia)

Actualmente existen varias empresas que ofertan soporte pagado en español:

E Global Services (Colombia) <http://grupoegs.com/koha/>
 Open Geek (Chile) <http://opengeekservice.cl/sitio/es/bibliotecas/koha.html>
 Orex (España) <http://www.orex.es/koha-soporte-y-mantenimiento/>
 Proyecto Koha Colombia <http://www.kohacolombia.org/>
 BiblioLatino (Perú) <http://www.bibliolatino.com/>
 SEK Soluciones (Venezuela) <http://sek.com.ve/Koha/>

4.6.2.2 PMB

PMB Services es la empresa que maneja en su mayoría servicios de soporte pagado, migración, capacitación, etc, en inglés y francés.

Adicionalmente en idioma español se ha podido identificar varias empresas que ofertan servicios con PMB:

Biteca (Colombia) <http://www.biteca.com/>
 Comunidad PMB (Colombia) <http://www.comunidadpmb.org/>

Factoría Thekes (España) <http://factoriathekes.es/pmb-para-bibliotecas/>

BiblioLatino (Perú) <http://www.bibliolatino.com/>

4.6.2.3 Calificaciones

Tabla No.27: Resumen calificaciones soporte

Criterio:	Soporte	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	9	Cuantitativamente Koha dobla las opciones de soporte de PMB, pero la disponibilidad de soporte en español es de 6 para Koha y 4 para PMB.
PMB	7	

Fuente: elaborado por el autor

4.7 Documentación

4.7.1 Koha

Adicionalmente al Wiki disponible en línea, este proyecto dispone de:

- Instrucciones de instalación del software
- Manual de uso de Koha hasta la versión 3.24 en inglés, tanto en versión web como pdf (la versión pdf consta de 902 páginas).
- Existen manuales de la versión 3.16 en inglés, chino, turco, checo y español.
- Varias versiones de mayor antigüedad hasta la versión 3.0
- En documentación para desarrolladores existen:
 - Guías para elaboración de código
 - Documentación del API de Koha
 - Esquema de la base de datos de Koha

4.7.2 PMB

PMB Services ofrece una versión impresa del manual a un costo, adicionalmente dispone de documentación en línea realizado en XML DocBook y dispone de los siguientes manuales:

- Guía de instalación en inglés, francés y español
- Manual de usuario en inglés, francés y español
- Manual del administrador en inglés y francés

4.7.3 Calificaciones

Tabla No.28: Resumen calificaciones documentación

Criterio:	Documentación	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	10	Koha dispone al menos de 5 manuales oficiales actualizados de documentación sobre varios temas mientras PMB dispone actualmente de 3.
PMB	7	

Fuente: elaborado por el autor

4.8 Seguridad

Revisando sitios web de reporte de vulnerabilidades desde 2015 encontramos lo siguiente:

4.8.1 Koha

A Junio de 2015 SBA Research encontró 6 vulnerabilidades en Koha ILS, tales como inyecciones SQL sin autenticación, inclusión de archivos locales, XSS y CRFS que podrían comprometer el programa y el servidor. Luego de publicarlas la comunidad de Koha corrigió estos problemas y realizó los security releases 3.16.12, 3.18.8 y 3.20.1.

En el sitio CVE Details (Common Vulnerabilities and Exposures) en 2015 reportó vulnerabilidades XSS en el módulo de Administración en las versiones 3.16.6 and 3.18.x hasta 3.18.2 permite ataques remotos mediante el argumento ordenar_por en el OPAC, solucionado en las versiones posteriores.

En Exploit Database se detalla la vulnerabilidad para la versión igual o menor a 3.20.1 por inyecciones SQL múltiples, solucionado en las versiones posteriores.

La agenda de liberaciones de Koha estipula liberaciones de versiones con corrección de bugs y seguridad mensualmente.

4.8.2 PMB

Las bases de datos de vulnerabilidades Exploit Database, CVE Details, SecuriTeam y del NIST (National Institute of Standards and Technology USA) detallan el CVE-2014-9457 a inicios de 2015. Es una vulnerabilidad a inyección SQL en classes/mono_display.class.php que permite a usuarios autenticados ejecutar comandos SQL arbitrarios a través del parámetro id en catalog.php. que afectó a la versión 4.1.3 y anteriores y que ya está solucionado.

En el histórico de versiones liberadas, en el año 2015 se realizaron 11 liberaciones de seguridad que es similar a la de Koha.

4.8.3 Calificaciones

Tabla No.29: Resumen calificaciones seguridad

Criterio:	Seguridad	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	8	La cantidad de vulnerabilidades documentadas en sitios especializados es similar y en ambos casos han sido resueltas en las nuevas versiones.
PMB	8	

Fuente: elaborado por el autor

4.9 Funcionalidad

Para una mejor valoración de las características funcionales de los dos proyectos se ha realizado una instalación completa en dos servidores independientes.

4.9.1 Koha

4.9.1.1 Módulo de Circulación

Posee opciones de préstamos, devoluciones, renovación, transferencia y catalogación rápida. Igualmente dispone de opciones de catalogación fuera de línea mediante archivo y descarga de aplicación de escritorio o extensión para Firefox. Los informes de circulación permiten ver las colas de reserva, preparación, reservas a retirar, proporciones, transferencias a recibir, retrasos y multas.

4.9.1.2 Módulo de Usuarios

Permite crear, buscar y filtrar usuarios en el sistema, tanto socios o usuarios de la biblioteca como staff administrativo, escuelas y bibliotecas propiamente dichas y completar todos los datos relevantes.

4.9.1.3 Módulo de Búsqueda Avanzada

Permite búsquedas compuestas tomando en cuenta palabras clave y operadores lógicos, tipo de ítem, límites de subtipo, ubicación, disponibilidad, y también limitar a una sola biblioteca, año o idioma dentro del sistema.

4.9.1.4 Módulo de Listas

Genera listas de recursos bibliográficos públicas y privadas para su uso en generar listas por

materia, curso, profesor, temáticas, bibliografías, syllabus, etc.

4.9.1.5 Módulo de Catalogación

Genera nuevos registros bibliográficos desde cero o con importación por Z39.50/SRU para su posterior modificación e inventariado.

4.9.1.6 Módulo de Autoridades

Se gestionan recursos uniformes normalizados para autoridades (autores, materias, editoriales y series), sean personas físicas, institucionales, compilaciones, etc., para que la catalogación a realizada genere nombres sin ambigüedades ni duplicaciones.

4.9.1.7 Módulo de Publicaciones Periódicas

Genera registros bibliográficos para publicaciones que se generar cada cierto período determinado de tiempo, gestionando la suscripción, clasificando por fascículos, por artículos (catalogación analítica) y con las opciones específicas para este tipo de recursos bibliográficos.

4.9.1.8 Módulo de Adquisiciones

Permite a las bibliotecas de tamaño mediano y grande que manejan internamente la gestión total o parcial de su presupuesto generar pedidos de recursos bibliográficos, definir un presupuesto, pedidos, administrar sugerencias de compra, facturas, administrar presupuestos activos e inactivos, manejar varios fondos de financiamiento y varias monedas dentro del módulo.

4.9.1.9 Módulo de informes

Genera informes guiados mediante un asistente, permite guardar y reusar informes, crear

nuevos a partir de SQL, dispone de un diccionario de informes que permite buscar los argumentos SQL en una lista de informes creados por la comunidad Koha y además dispone de un esquema de la base de datos de Koha para poder desarrollar nuevos informes.

Además la opción de asistentes de estadísticas permite generar datos con varios atributos dentro de adquisiciones, usuarios, catálogo, circulación, publicaciones periódicas y reservas.

La sección de listas principales destaca los usuarios con más préstamos y los ítems más presatados, al contrario que los inactivos permiten identificar usuarios que no han hecho préstamos e ítems nunca prestados y otros como ítems perdidos, catálogo por tipo de ítem.

4.9.1.10 Módulo Herramientas

Son varios aplicativos para actividades variadas que se encuentran divididos en tres áreas:

4.9.1.10.1 Usuarios y circulación

Administrar listas, comentarios, importación de datos de usuarios, definir avisos y recibos, aviso de retraso, crear carnés de usuarios, modificación o eliminación en lote, moderar etiquetas, cargar imágenes de usuario.

4.9.1.10.2 Catálogo

Eliminación o modificación de ítems o registros en lote, exportación de datos, inventarios/ existencias, creación de etiquetas y etiquetas en columnas, administrar colecciones rotativas, plantillas MARC personalizadas, importación y administración de registros MARC, carga de imágenes de cubiertas para el OPAC.

4.9.1.10.3 Herramientas adicionales

Manejo de calendario de atención de la biblioteca, perfildes de exportación CSV, visor de log

del sistema, noticias para el OPAC e interfaz administrativa, planificador de tareas, frases del día.

4.9.1.10.4 Módulo de Administración de Koha

Este módulo posee una considerable cantidad de opciones para parametrizar prácticamente la totalidad del sistema de gestión de bibliotecas y está dividido en preferencias globales del sistema, parámetros básicos, usuarios y circulación, plugins, catálogo, parámetros de la adquisición, parámetros adicionales.

4.9.1.10.5 Módulo Acerca de Koha

Este apartado informativo permite conocer gran parte de los detalles de información del servidor, módulos de Perl, información del sistema, equipo Koha, licencias, traducciones y línea de tiempo del proyecto Koha.

4.9.2 PMB

4.9.2.1 Módulo de Circulación

Maneja préstamos, devoluciones, documentos a tratar, grupos de usuarios, nuevos usuarios y periódicos.

4.9.2.1.1 Cestas

Gestión de cestas, procedimientos y procedimientos remotos. Recolección de cestas por rama, código de barras y selección. Acciones para vaciar, transferir, ediciones, mailing, procesos y eliminar cestas.

4.9.2.1.2 Reservas , Sugerencias y Caducados

Reservas actuales y pasadas, validación de documentos a colocar. Generación de nueva sugerencia

4.9.2.2 Módulo de Catálogo

Permite búsquedas por varios tipos de conceptos, de tipo avanzado y consulta externa, registro de nuevos documentos bibliográficos y gestión de publicaciones periódicas, manejo de cestas y estantes virtuales, como consultas, conecciones Z39.60 y manejo de sugerencias bibliográficas.

4.9.2.3 Módulo de Autoridades

Maneja la normalización en autores, categorías, editoriales, colecciones y subcolecciones, títulos seriados y uniformes, así como clasificaciones. Además el submódulo de semántica permite manejar sinónimos y la importación de listas de autoridades.

4.9.2.4 Módulo de Informes

Dispone de estadísticas y listas personalizables clasificadas de ejemplares, lectores, préstamos, además parametrizar y generar informes propios parametrizables. Adicionalmente de informes por ejemplares, reservas, usuarios, publicaciones periódicas, generación de códigos de barras y de noticias para el OPAC.

4.9.2.5 Módulo de Diseminación Selectiva de la Información (DSI)

Este módulo es uno de los elementos diferenciadores de PMB sobre la mayoría de programas de gestión integral de bibliotecas, pues permite mantener informados por correo de las novedades generales y específicas a la totalidad o grupos de usuarios registrados de las

bibliotecas. Aquí existen opciones para su difusión automática o manual, cestas públicas y privadas y definición de fuentes RSS.

4.9.2.6 Módulo de Administración

Este módulo contiene una gran cantidad de parámetros para el manejo del sistema de gestión que comprende mayormente ejemplares, registros, usuarios, administradores, herramientas, OPAC, acciones, módulos, adquisiciones y parámetros de administrador.

4.9.3 Calificaciones

Tabla No.30: Resumen calificaciones funcionalidad

Criterio:	Funcionalidad	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	9	Las características funcionales de ambos programas es muy similar. Desde el punto de vista bibliotecológico KOHA tiene la ventaja de poseer el formato de campos MARC 21 completo mientras que PMB posee formato MARC 21 abreviado, lo cual permite a Koha un nivel de catalogación con granularidad más amplia.
PMB	8	Por otro lado PMB se destaca en que posee un módulo de diseminación selectiva de la información (DSI) que es de gran ayuda para actividades de promoción de los servicios de las bibliotecas.

Fuente: elaborado por el autor

4.10 Integración

4.10.1 Modularidad

Koha permite el uso de plugins (desabilitados por defecto como medida de seguridad) como módulos adicionales que pueden ser desarrollados, e igualmente en su comunidad existe documentación acerca de como desarrollar estos aditamentos.

PMB no dispone de una modularidad en una forma tan clara lo que hace más difícil su implementación.

4.10.2 Estándares

Tabla No.31: Comparativa estándares Koha y PMB

Estandar	Koha	PMB
Cliente Z39.50	si	si
Consultas SPARQL	no	si
Digital document viewer	no	si
Django templates	no	si
EDM WebDAV server	no	si
Indexación de georeferenciados	no	si
Integración con Zotero	si	si
Interface LDAP (AD)	si	si
Interfaz RFID	no, funcionalidad solicitada	si
ISO 25964 para manejo de tesauros	no	si
ISO 2709 de intercambio de información bibliográfica	si	si
OAI import/export	solo como servidor y no harvesting	si
OPAC 2.0	si	si

Estandar	Koha	PMB
OpenSearch	si	no
OpenURL (url con metadatos para bibliotecas)	si	si
Procesado voz a texto de archivos de audio y video para búsqueda de texto completo	no	si
RDA	si	si
RDF	no, en proyecto SemantiKoha	si
RSS feeds	si	si
Servidor z39.50	si	no
SRU/SRW	si	si
Unimarc	si	si
Uso de Amazon Books Webservices	si	si
Web 2.0	si	si
Web services	si ILS-DI (discovery interface)	si

Fuente: elaborado por el autor

4.10.3 Compatibilidad con otras aplicaciones

PMB maneja visor para desplegar archivos de varios tipos como pdf, word, jpg, etc. En lo referente a autenticación ambos programas trabajan con LDAP, mientras que en soporte a interfaz a RFID es nativo en PMB y no en Koha. Anteriormente existía un API de Koha para el CMS Drupal que no ha sido modificado por más de 2 años. Es de destacar la implementación de RDF en PMB para poder generar consultas SparQL.

4.10.3.1 Requerimientos del software

Ambos sistemas trabajan básicamente un servidor LAMP (Linux Apache MySQL PHP) y se pueden manejar adecuadamente en ambientes de virtualización como Oracle VirtualBox,

VMWare, etc. y en contenerización como Docker.

4.10.4 Calificaciones

Tabla No.32: Resumen calificaciones integración

Criterio:	Integración	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	8	Las diferencias en este aspecto no son notorias, si bien por un lado KOHA se destaca en que puede ser servidor del protocolo Z3950 mientras que solo PMB actualmente tiene soporte para RDF.
PMB	8	

Fuente: elaborado por el autor

4.11 Origen y objetivos

Ambos proyectos, si bien poseen similitudes en muchos aspectos, tienen igualmente rasgos diferenciadores desde sus inicios que han influido en su desarrollo, reflejan claramente idiosincrasias dominantes distintas, la francesa y neozelandesa, utilizan lenguajes distintos, PHP (que inició en 1995) frente a Perl (apareció en 1987).

De igual forma, en el génesis de Koha en 1999 intervinieron instituciones como el Horowhenua Library Trust y Katipo Communications, lo que le dió un enfoque corporativo mientras que PMB fue creado por un bibliotecario, François Lemarchand que en 2002 creó la primera versión y que luego se le unieron paulatinamente otros colaboradores de las siguientes versiones iniciales fundaron la empresa PMB Services que sigue manteniendo el grueso del desarrollo del proyecto.

De estas aproximaciones se puede decir que Koha nació como un ILS para bibliotecas grandes o muy grandes que luego en su evolución se ha pensado también en centros de documentación medianos y pequeños, mientras que el enfoque de PMB es el opuesto, iniciando con bibliotecas

medias y evolucionando a su utilización en bibliotecas grandes. Incluso se podría inferir que PMB es un buen candidato para su uso en bibliotecas universitarias que se encuentren en sus primeras etapas de desarrollo mientras que KOHA es un sistema más adecuado para centros con mayor infraestructura, número de personal y presupuesto.

En los objetivos existen coincidencias en el enfoque hacia las libertades pregonadas por el software libre en los dos proyectos, y sus características comunes y diferenciadoras no hacen más que enriquecer de una forma extraordinaria la oferta de sistemas de gestión integral de bibliotecas.

4.11.1 Calificaciones

Tabla No.33: Resumen calificaciones origen y objetivos

Criterio:	Origen y objetivos	
<i>Software</i>	<i>Valor sobre 10</i>	<i>Notas</i>
Koha	9	Ambos proyectos son similares en su valoración, aunque sus inicios e idiosincrasias son distintos, los dos tienen un origen y objetivos en razonable consonancia con lo que esta evaluación busca.
PMB	9	

Fuente: elaborado por el autor

4.12 Resultados

Una vez realizadas las valoraciones de los distintos criterios las evaluaciones muestran resultados claros, ya que Koha presenta mejores puntuaciones en todos los criterios donde no existe empate.

Los criterios donde existe la misma valoración son: longevidad, nuevas tecnologías, licencias, seguridad, integración y origen y objetivos que poseen un peso específico medio o bajo. Los pesos específicos de cada criterio están también reflejados en esta tabla.

Tabla No.34: Resultados de evaluación de Koha y PMB

Criterio	Peso específico	KOHA		PMB	
		Valor	Ponderado	Valor	Ponderado
Comunidad	10	9	9	6	6
Liberación de versiones	8	10	8	6	4,8
Longevidad	6	10	6	10	6
Nuevas tecnologías	4	8	3,2	8	3,2
Licencias	4	10	4	10	4
Soporte	6	9	5,4	7	4,2
Documentación	6	10	6	7	4,2
Seguridad	4	8	3,2	8	3,2
Funcionalidad	8	9	7,2	8	6,4
Integración	6	8	4,8	8	4,8
Origen y objetivos	2	9	1,8	9	1,8
Total ponderado			58,6		48,6

Fuente: elaborado por el autor

El peso específico dado ha sido estimado en números pares buscando privilegiar los aspectos más importantes en este caso particular con énfasis en criterios de comunidad, funcionalidad y versiones para asegurar seleccionar un proyecto que privilegie justamente esos aspectos y poniendo en un segundo grupo de importancia la longevidad, soporte, documentación e integración.

En general se puede afirmar que Koha cumple con mejores características como una solución de sistema integrado de gestión de bibliotecas para las instituciones universitarias del Ecuador en especial por su amplia y participativa comunidad, que es base para un proyecto de software libre exitoso en el tiempo, funcionalidades completas y robustas así como la liberación periódica y planificada de versiones.

4.13 Conclusiones del caso de estudio

Para el presente caso se investigó y encontró 22 proyectos viables (tabla No.11), de los mismos se seleccionó seis que cumplían las características mínimas establecidas para este caso.

De los seis candidatos de la tabla No. 21 y realizada una selección sobre 4 criterios (Funcionalidad, comunidad, liberación de versiones y longevidad) se obtuvo dos candidatos finales (Koha y PMB) sobre los cuales se realizó una evaluación exhaustiva la cual nos ha dado los resultados de la tabla No. 22 que destaca a Koha como el proyecto que más se adecúa a las necesidades del caso.

4.14 Análisis de validez de los resultados obtenidos

A partir de los resultados obtenidos al final del proceso de evaluación de software para bibliotecas universitarias en el país se busca una validación de los mismos a través de una prueba práctica de instalación y puesta en funcionamiento de los programas finalistas así como con la búsqueda de evaluaciones similares publicadas.

4.14.1 Prueba de funcionalidad en importación de registros y rendimiento general

Para comprobar en la práctica la adecuación de ambas aplicaciones en tareas y flujo de trabajo de una biblioteca universitaria se procedió a realizar una exportación de registros bibliográficos realizados en SIABUC 9 (software ILS propietario que es usado en gran parte de las bibliotecas universitarias en el país).

La importación se hizo mediante el protocolo ISO 2709 de 6.712 registros bibliográficos (fichas) de libros con catalogación técnica de tercer nivel y tomó 9 minutos generar el archivo formato .mrc (encoding utf-8 windows crlf)

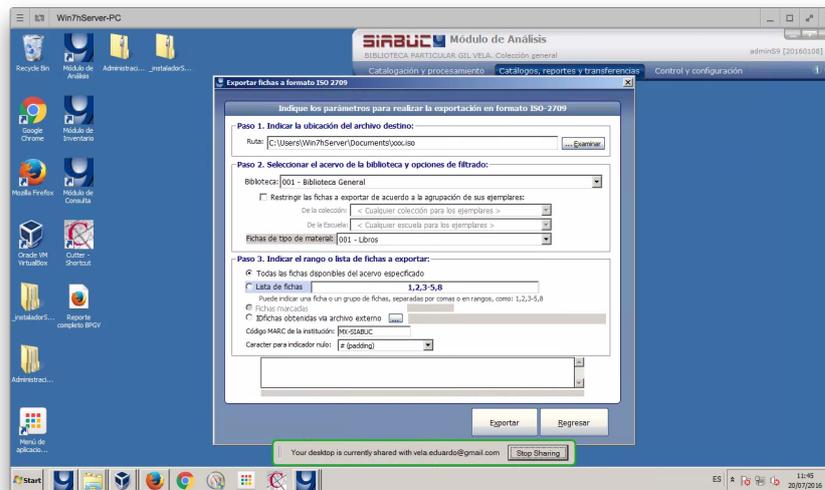


Figura 5 Importación de registros bibliográficos desde el software SIABUC 9

Fuente: elaborado por el autor

4.14.1.1 Configuración de servidor de pruebas

Para la prueba de trabajo de Koha y PMB se utilizó un computador portátil con las siguientes características principales:

- Procesador: Intel Core i7-2640M (SandyBridge)
2.80Ghz, 4MB L3 Cache
Dos núcleos, cuatro subprocesos
- Memoria: 8GB DDR3
- Almacenamiento: SSD de 250 gb. 6Gb/s SATA
- Redes: Ethernet 10/100/1000
802.11b/g/n Wireless LAN
- Sistema Operativo: Ubuntu Linux 16.04 LTS 64 bits

Sobre este equipo se instaló VirtualBox v. 5.0.18_Ubuntu y se crearon dos máquinas virtuales

idénticas con las siguientes características para instalar y configurar tanto Koha como PMB:

- Memoria: 2 gb.
- Disco duro 10 gb.
- Red: Adaptador puente asignado un IP fijo
- Sistema Operativo: Ubuntu Server 14.04.4 LTS con stack LAMP

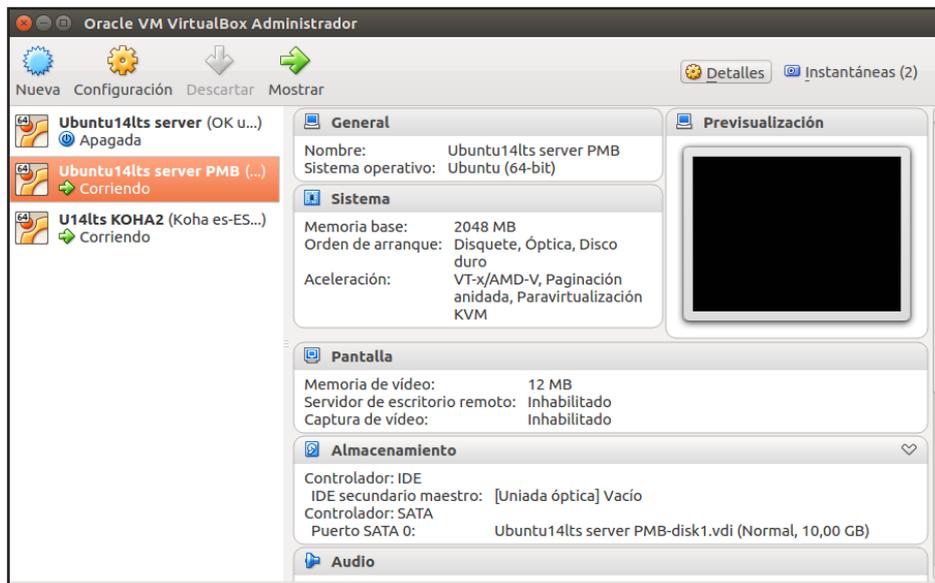


Figura 6 Servidor configurado con 2 máquinas virtuales en VirtualBox

Fuente: elaborado por el autor

4.14.1.2 Instalación y configuración de los paquetes de software

4.14.1.2.1 Instalación de KOHA

Se realizó instalación de ssh para acceso remoto

Se configuró el repositorio de Koha en la lista de fuentes de Ubuntu

Instalación de Koha y dependencias

Configuración acceso web con dirección IP y puerto de servicio 8080

Seteo de Apache y base de datos koha001

Password xUPVT42iZmyMZaDI

Username koha_koha001

El instalador no incluye el lenguaje español por lo que se procede a instalar los archivos de traducción es-ES previos a la configuración

La versión instalada es Koha 16.05.01.000

Database settings:

database type : mysql

database name : koha_koha001

database host : localhost

database port : 3306

database user : koha_koha001



Figura 7 Instalación y configuración de Koha

Fuente: elaborado por el autor

Tablas creadas en la base de datos exitosamente

Se selecciona Marc21 como formato a usar sobre Unimarc

Datos obligatorios ingresados en SQL

Se ingresa por el puerto de servicio 8080, login, inicialmente se cambia el lenguaje en el panel

de preferencias I18N/L10N y se establece el formato de fecha en día/mes/año que es el que se acostumbra en el país. Finalmente se crea una primera biblioteca (001)



Figura 8 Interface de administración de Koha instalada

Fuente: elaborado por el autor

4.14.1.3 Instalación PMB

Se asigna el ip 192.168.0.152

Se realizó instalación de ssh para acceso remoto

Se descarga la más reciente versión estable PMB 4.2.1

En mysql se crea la base de datos pmb001

Se asigna permisos al usuario pmb001 con la clave .pmb1740

Se accede al instalador, se elige codificación iso-8859-1, lenguaje español y clasificación Dewey



Figura 9 Instalación de PMB

Fuente: elaborado por el autor

Una vez configurada la aplicación se actualiza la estructura de la base de datos de v5.18 a v5.19 en el panel de administración>herramientas>actualización de la base de datos de PMB.

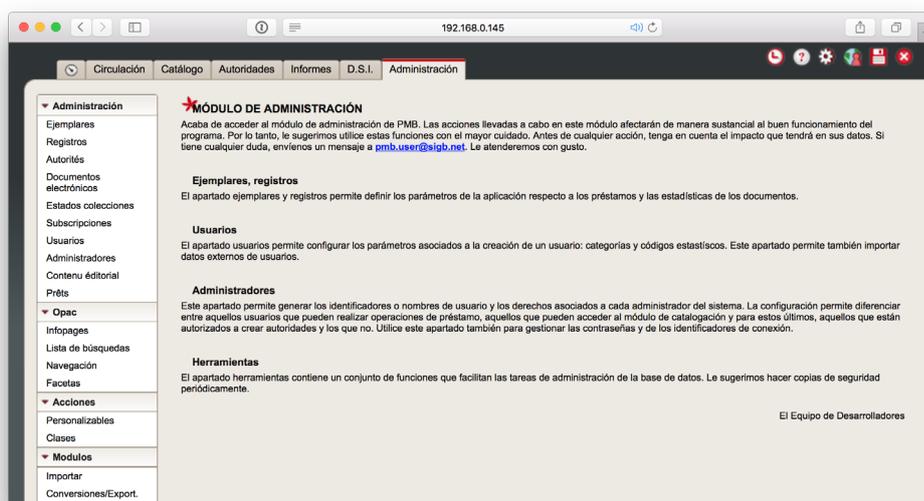


Figura 10 Interface administrativa de PMB funcionando

Fuente: elaborado por el autor

4.14.1.4 Migración de registros desde SIABUC 9

4.14.1.4.1 KOHA

Se ingresa a Herramientas > Preparar registros MARC para importación y se carga el archivo exportado de SIABUC 9

Registros preparados MARC para importar

- Seleccione un archivo de registros MARC para importar en el depósito. Será procesado, y cada registro válido analizado para su posterior importación en el catálogo.
- Puede introducir un nombre para esta importación. Puede ser útil, cuando se crea un registro, para recordar de dónde vienen los datos de MARC sugeridos!

Importar registros dentro en el depósito

Seleccione el archivo a preparar:

Porcentaje subido:

Comentarios acerca de este archivo:

Tipo de registro:

Codificación de caracteres:

¿Buscar los registros existentes en el catálogo?

Regla de coincidencia de registro:

Acción en caso de encontrar registro coincidente:

Acción si no se encontró coincidencia:

¿Verificar datos de ejemplar embebidos?

Sí

No

Como procesar los ítems:

Figura 11 Preparación para importación de registros a Koha

Fuente: elaborado por el autor

La preparación de importación despliega el resumen:

Resultados de preparación MARC:

Procesando registros bibliográficos

6712 registros en el archivo

0 registros no importados debido a error en el formato MARC

6712 registros importados

No se comprobaron coincidencias con los registros existentes en el catálogo

0 items procesados

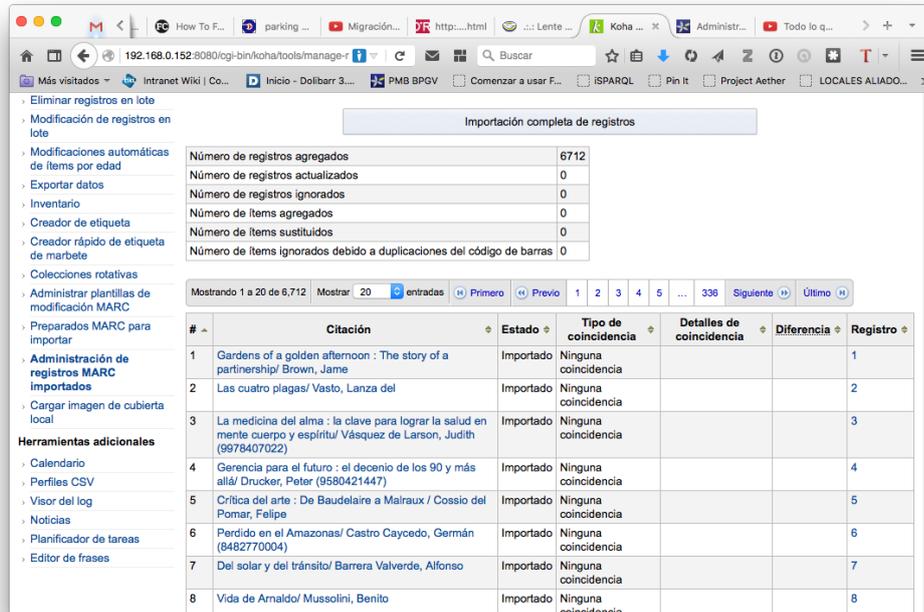


Figura 12 Confirmación de importación de registros a Koha

Fuente: elaborado por el autor

Se procesaron correctamente los 6712 registros importados, es decir el 100% de los registros existentes en el archivo ISO.

4.14.1.4.2 PMB

Se procede a realizar la transformación del archivo de registros mediante la opción Herramientas > Conversiones/Export. > Conversiones de archivos externos y se escoge la opción USMARC to UniMARC

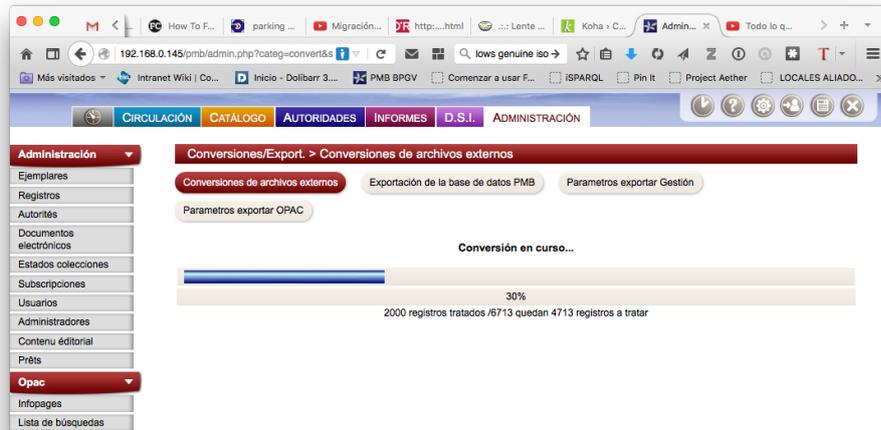


Figura 13 Importación de registros a PMB

Fuente: elaborado por el autor

La información SQL de PMB arroja los siguientes resultados luego de la importación:

Número total de registros	5767
Número total de ejemplares	0
Número total de boletines	0
Número total de autores	4605
Número total de editores	2987
Número total de prestatarios	0
Número total de préstamos en curso	0
Número total de préstamos en curso y archivados	0

Una vez realizada la transformación se procede a la importación de los registros el cual nos arroja un total de 5767 registros de los 6712 del archivo de importación, lo que da un resultado de 85% de efectividad en la importación ISO.

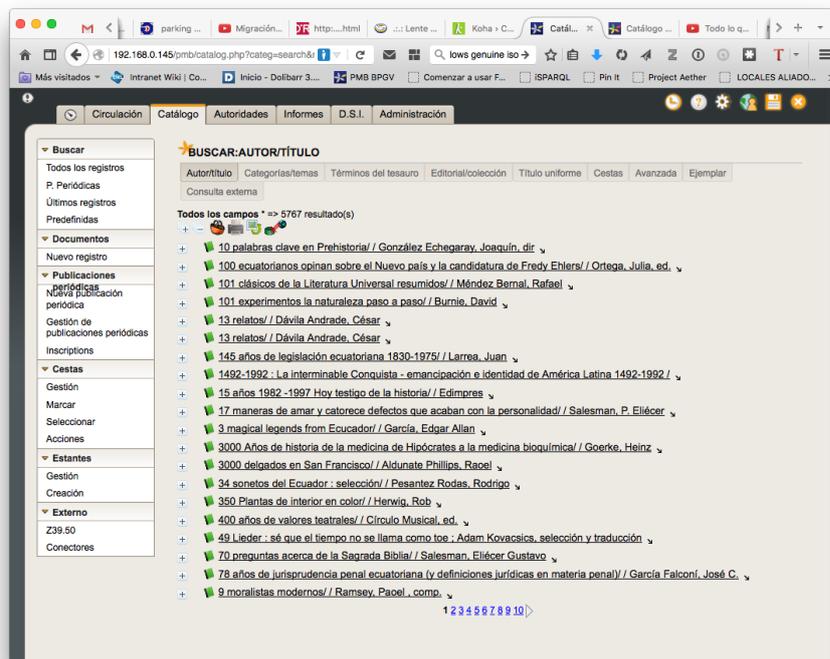


Figura 14 Registros importados disponibles en PMB

Fuente: elaborado por el autor

4.14.1.5 Formato MARC21 / ISO 2709 para intercambio de información bibliográfica

Este es el formato elegido para la exportación de datos bibliográficos desde un sistema en funcionamiento, en este caso SIABUC 9, a KOHA y PMB.

A continuación se realiza una comparación de un registro dentro del archivo de exportación MARC21 y su correspondiente visualización una vez importado en KOHA y PMB

Tabla No.35: Visualización de registro MARC21

Formato													
Archivo .mrc	<p>0450000100050000000300010000504000050000602000200001108200150003124501110004 6260004300157300002200200300001000222505049900232650002300731650001900754650 0026007736500019007996500028008186500020008464466## a## a84-494-1797-X 0# a940 H673 0# aHistoria del mundo moderno : entre el renacimiento y la revoluci�n francesa/ cCar- los Gispert, direcci�n 4# aBarcelona : bOc�ano Grupo Editorial 4# a3 tms. / 30 cm. 4# bil. 4# aTm.1: Renacimiento, Barroco, ilustraci�n, t: 2: Entre la revoluci�n industrial y el colonialismo: el siglo XIX progreso y revoluci�n, De la sociedad rural a la sociedad industrial, La cultura del siglo XIX, Africa y Ocean�a, tm. 3: Entre la gran guerra y el nuevo milenio: coonflictos y guerras, El desarrollo de la ciencia y la t�cnica, Poblaci�n, ideolog�as y movimientos sociales, pensamiento y religi�n, La cultura en el siglo XX. Los dea�os de la globalizaci�n, El regreso de la historia ## aHistoria moderna ## aRenacimiento ## aRevoluci�n francesa ## alperialismo ## alndustria z-historia ## aGlobalizaci�n</p>												
KOHA	<p>Historia del mundo moderno : entre el renacimiento y la revoluci�n francesa/ Carlos Gispert, direcci�n - Barcelona : Oc�ano Grupo Editoria - 3 tms. / 30 cm il</p> <p>Tm.1: Renacimiento, Barroco, ilustraci�n, t: 2: Entre la revoluci�n industrial y el colonialismo: el siglo XIX progreso y revoluci�n, De la sociedad rural a la sociedad industrial, La cultura del siglo XIX, Africa y Ocean�a, tm. 3: Entre la gran guerra y el nuevo milenio: coonflictos y guerras, El desarrollo de la ciencia y la t�cnica, Poblaci�n, ideolog�as y movimientos sociales, pensamiento y religi�n, La cultura en el siglo XX. Los dea�os de la globalizaci�n, El regreso de</p> <p>ISBN: 84-494-1797-X</p> <p>Subjects--Topical Terms:</p> <p>Histo R Revoluci</p> <p>Industria G---historia ##</p> <p>Dewey Class. No.: 940 H673</p>												
PMB	<p>P�blico ISBD</p> <p>Historia del mundo moderno : entre el renacimiento y la revoluci�n francesa/ [texto impreso] . - Barcelona . : Oc�ano Grupo Editorial. [s.d.] . - 3 tms. / 30 cm. : il. ISBN : 978-84-494-1797-9</p> <p>Etiquetas: Historia moderna Renacimiento Revoluci�n francesa Imperialismo Industria -historia Globalizaci�n</p> <p>Clasificaci�n: 940 H</p> <p>Nota de contenido: Tm.1: Renacimiento, Barroco, ilustraci�n, t: 2: Entre la revoluci�n industrial y el colonialismo: el siglo XIX progreso y revoluci�n, De la sociedad rural a la sociedad industrial, La cultura del siglo XIX, Africa y Ocean�a, tm. 3: Entre la gran guerra y el nuevo milenio: coonflictos y guerras, El desarrollo de la ciencia y la t�cnica, Poblaci�n, ideolog�as y movimientos sociales, pensamiento y religi�n, La cultura en el siglo XX. Los dea�os de la globalizaci�n, El regreso de la historia</p> <p>Ejemplares</p> <table border="1"> <thead> <tr> <th data-bbox="411 1921 515 1944">C�digo de barras</th> <th data-bbox="675 1921 738 1944">Signatura</th> <th data-bbox="826 1921 914 1944">Tipo de medio</th> <th data-bbox="1042 1921 1106 1944">Ubicaci�n</th> <th data-bbox="1193 1921 1257 1944">Secci�n</th> <th data-bbox="1321 1921 1369 1944">Estado</th> </tr> </thead> <tbody> <tr> <td data-bbox="411 1944 515 1966">ning�n ejemplar</td> <td></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	C�digo de barras	Signatura	Tipo de medio	Ubicaci�n	Secci�n	Estado	ning�n ejemplar					
C�digo de barras	Signatura	Tipo de medio	Ubicaci�n	Secci�n	Estado								
ning�n ejemplar													

Fuente: elaborado por el autor

4.14.1.6 Comparación de resultados de búsquedas en Koha y PMB

Como prueba de búsqueda se procede a buscar en ambos programas con los términos “ciclismo urbano” que en ambos casos nos arroja un resultado como ilustra la figura de abajo.

The figure consists of two screenshots. The top screenshot shows the Koha search results for the book 'Manual de ciclismo urbano / Secretaría de Movilidad Municipio del Distrito Metropolitano'. It includes details such as the editor (Dirección Metropolitana de Deporte y Recreación), description (39 p. / 20 cm.), ISBN ([Pendiente]), and classification (796.6). The bottom screenshot shows the PMB search results for the same book, displaying a table of metadata and a table of copies (Ejemplares) which currently shows 'ningún ejemplar'.

Koha Search Results:

- Vista normal | Vista MARC | Vista ISBD
- Manual de ciclismo urbano / Secretaría de Movilidad Municipio del Distrito Metropolitano**
- Editor: Quito, Ecuador : Dirección Metropolitana de Deporte y Recreación
- Descripción: 39 p. / 20 cm.
- ISBN: [Pendiente]
- Materia(s): 1. Ciclism | 2. Bicicleta
- Clasificación CDD: 796.6
- Contenidos:
1. La bicicleta - 2. Mantenimiento de la bici - 2. Ciclismo urbano - 3
- Etiquetas de esta biblioteca: No hay etiquetas de esta biblioteca para este título. [Agregar etiquetas](#)
- ★★★★★ valoración media: 0.0 (0 votos)

PMB Search Results:

Público | ISBD

Título :	Manual de ciclismo urbano /
Tipo de documento:	texto impreso
Editorial:	Quito, Ecuador : : Dirección Metropolitana de Deporte y Recreación
Número de páginas:	39 p. / 20 cm.
ISBN/ISSN/DL:	[Pendiente]
Etiquetas:	1. Ciclismo 2. Bicicletas
Clasificación:	796.6
Nota de contenido:	1. La bicicleta - 2. Mantenimiento de la bici - 2. Ciclismo urbano - 3.

Ejemplares

Código de barras	Signatura	Tipo de medio	Ubicación	Sección	Estado
ningún ejemplar					

Figura 15 Pantallas de resultado de búsqueda de Koha (superior) y PMB (inferior)

Fuente: elaborado por el autor

Los tiempos de respuesta para esta búsqueda son de 5,2 segundos para Koha y 2,6 segundos para PMB, y con otras pruebas para distintas búsquedas con mayor complejidad la relación de tiempo de búsqueda es similar.

Como referencia externa, SIABUC 9 (basado en PostgreSQL y Vbasic .NET) realizó la misma búsqueda en los datos originales de la importación en 8,02 segundos, pero cabe anotar que

este software funciona en un servidor distinto, con diferente configuración por restricciones de su licencia por lo que los datos cuantitativos no son comparables para efectos de rendimiento.

4.14.2 Contrastación de resultados con evaluaciones similares

Una vez conocidos los resultados del presente trabajo se ha buscado casos de evaluaciones similares para poder contrastar los mismos. (Müller, 2011) ha realizado un estudio sobre software para gestión integral de bibliotecas para la Fundación para una Biblioteca Global de Quebec, Canadá, y a través de una metodología diferente y tomando en cuenta que el enfoque de público objetivo es distinto ha llegado a una conclusión similar que Koha es la mejor alternativa y dejando a PMB y Evergreen como los otros finalistas como el siguiente gráfico ilustra.

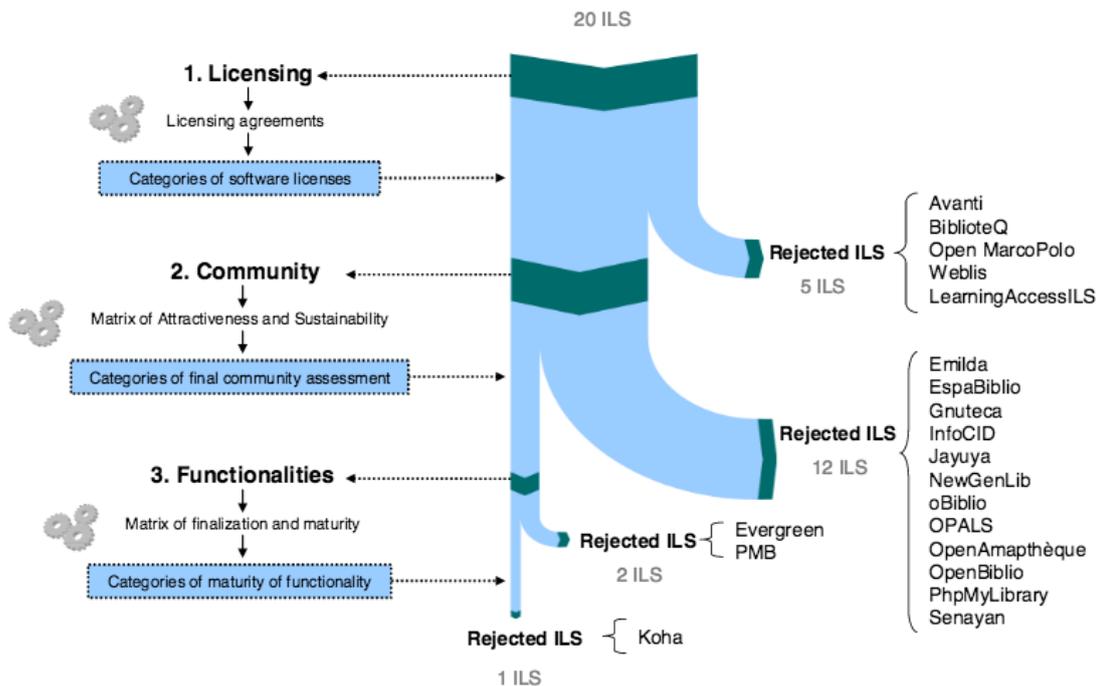


Figura 16 Resultados de evaluación de Muller

Tomado de (Müller, 2011)

CONCLUSIONES

A la terminación de este trabajo se concluye que:

Tanto la razón de ser de las bibliotecas públicas como del software libre confluyen en el ideal de compartir el conocimiento libremente por lo que su implementación conjunta además de razones de economía y eficiencia también posee coherencia en sus principios.

Analizados varios métodos de evaluación de software libre se escogió la metodología Van den Berg por sus características, actualidad y capacidad de adecuación a las necesidades de la evaluación a realizar.

Se desarrolló a lo largo de este trabajo un modelo de evaluación de software con adecuaciones hacia el software de gestión integral de bibliotecas.

A lo largo de la presente investigación se ha podido comprobar que existen varias alternativas de software libre viables para su uso en la gestión de bibliotecas universitarias en el Ecuador, en especial PMB y KOHA, siendo este último el que mejores ventajas presenta para su implementación y uso.

Una vez realizada la evaluación, para mejor comprobación de sus resultados se elaboró una prueba de funcionalidad práctica de los dos candidatos finales para efectos de comprobación del proceso realizado, la cual fue satisfactoria.

En las bibliotecas universitarias ecuatorianas existen pocas experiencias de uso e implementación de software libre para la gestión integral de bibliotecas. La principal fuente de experiencia en implementación de este tipo de software puede ser el de la Biblioteca Nacional Eugenio Espejo de la Casa de la Cultura Ecuatoriana con Koha.

Actualmente existe dentro de las universidades ecuatorianas la infraestructura para la

implementación de programas de gestión integral de bibliotecas así como la capacidad técnica por parte de sus respectivos departamentos de sistemas informáticos.

Las instituciones y normas que rigen el funcionamiento de las universidades en este país establecen la necesidad de contar con un software de gestión bibliotecaria y este puede ser de tipo software libre.

El análisis comparativo, la selección y evaluación de software libre para gestión de bibliotecas realizado en el presente trabajo nos permite aseverar que existe en este campo proyectos disponibles con la madurez y funcionalidad adecuadas, ya que hay al menos 2 proyectos que pueden ser implementados en el 100% de las bibliotecas universitarias del país de ser necesario ya que disponen de las características funcionales necesarias, así como de la capacidad de manejar el volumen de información catalogada que disponen las bibliotecas universitarias actualmente conforme el anexo B.

La evaluación realizada en el presente trabajo es una guía de referencia como punto de partida para eventuales procesos de toma de decisiones sobre implementación de software libre de gestión bibliotecaria en las universidades ecuatorianas.

RECOMENDACIONES

La implementación de soluciones de software libre para gestión de bibliotecas universitarias en el país es posible y requiere de un acompañamiento constante de responsables del área informática en especial en las primeras fases de implementación de las mismas.

Los programas candidatos finales son soluciones cliente servidor que abarcan gran cantidad de módulos y funcionalidades básicas y avanzadas que requieren de las personas responsables de los centros bibliotecarios de un aprendizaje y experimentación constante por lo que es recomendable la existencia de tanto un servidor de trabajo y producción como uno de pruebas donde se pueda experimentar nuevas funcionalidades y flujos de trabajo.

Para un mejor avance en la curva de aprendizaje durante la implementación de Koha es recomendable unirse o conformar una comunidad de usuarios en situaciones similares, disponer de proveedores de soporte y asignar personal técnico tanto en la parte informática como bibliotecológica, preferiblemente con dominio de idioma inglés, para que recopilen y procesen la extensa documentación existente.

La migración de datos de otro software hacia Koha o PMB es posible pero siempre es recomendable realizarla con un posterior inventario físico de ejemplares ya que el sistema de gestión bibliotecaria se constituye en la base del registro de los bienes bibliográficos de un centro por lo cual un cambio de este tipo requiere este tipo de procedimiento.

El establecer un grupo de trabajo interinstitucional que permita compartir experiencias en el uso de Koha o PMB en varios centros bibliotecarios universitarios en el país permitiría compartir recursos de aprendizaje, experiencias y buenas prácticas que permitan una mejor y más eficiente implementación de los mismos.

BIBLIOGRAFÍA

Ågerfalk, P. J., Boldyreff, C., González-Barahona, J. M., Madey, G. R., & Noll, J. (2010). Open Source Software: New Horizons: 6th International IFIP WG 2.13 Conference on Open Source Systems, OSS 2010, Notre Dame, IN, USA, May 30 - June 2, 2010, Proceedings. Springer Science & Business Media.

Andrade, I. (2011). «Propuesta de un modelo de gestión de conocimiento para las organizaciones basado en herramientas Web 2.0: Caso Biblioteca “Benjamín Carrión” de la UTPL» (Thesis). Recuperado a partir de <http://dspace.utpl.edu.ec/jspui/handle/123456789/5481>

Anuradha, K. T., Sivakaminathan, R., & Kumar, P. A. (2011). Open-source tools for enhancing full-text searching of OPACs: Use of Koha, Greenstone and Fedora. Program: electronic library and information systems, 45(2), 231-239. <http://doi.org/10.1108/00330331111129750>

Arjona, L. (2012). What happened to OpenBRR (Business Readiness Rating for Open Source)? Recuperado a partir de <https://larjona.wordpress.com/2012/01/06/what-happened-to-openbrr-business-readiness-rating-for-open-source/>

Arriola Navarrete, Ó., & Butrón Yáñez, K. (2008). Sistemas integrales para la automatización de bibliotecas basados en software libre. ACIMED, 18(6), 0-0.

Arroyo Fernández, D. (2012). Implementación de Koha-Kobli en las bibliotecas de la Administración General del Estado, 2011-2012 : Informe final. Recuperado a partir de <http://travesia.mcu.es/portalnb/jspui/handle/10421/6522>

ASLE. (2014). Filosofía | ASLE. Recuperado a partir de <http://www.asle.ec/filosofia/>

Atos. (2004). Qualification and Selection of Open Source software (QSOS). Recuperado 26 de enero de 2015, a partir de <http://www.qsos.org/Method.html>

Best of Opensource.com: Education. (s. f.). Recuperado 3 de enero de 2016, a partir de <https://opensource.com/education/15/12/best-education-2015>

Bissels, G. (2008). Implementation of an open source library management system: Experiences with Koha 3.0 at the Royal London Homoeopathic Hospital. *Program: electronic library and information systems*, 42(3), 303-314. <http://doi.org/10.1108/00330330810892703>

Bissels, G., & Chandler, A. (2010). Two years on: Koha 3.0 in use at the CAMLIS library, Royal London Homoeopathic Hospital. *Program: electronic library and information systems*, 44(3), 283-290. <http://doi.org/10.1108/00330331011064276>

B., N., S., D., R., B., & S., S. R. (2010). Implementation of Automated Library Management System in the School of Chemistry Bharathidasan University using Koha Open Source Software. *International Journal of Applied Engineering Research*, Dindigul, 1(2), 149-167.

Breeding, M. (2009). The viability of open source ILS. *Bulletin of the American Society for Information Science and Technology*, 35(2), 20–25. <http://doi.org/10.1002/bult.2008.1720350207>

Benoit, B. (2005). Analyse technique des SIGB Koha et PMB. Recuperado 25 de mayo de 2016, a partir de <http://orbi.ulg.ac.be/handle/2268/135800>

Buchanan, K., & Krasnoff, B. (2005). Can Open Source Software Save School Libraries Time and Money? *Knowledge Quest*, 33(3), 32-34.

Café, L., Santos, C. dos, & Macedo, F. (2001). Proposta de um método para escolha de software de automação de bibliotecas. *Ciência da Informação*, Brasília, 30(2), 70–79.

Calvo Lázaro, M., Romay Sieira, J., Piñeiro Vazquez, J. V., Alonso Martín, I., Quintáns Míguez, J. E., & Gijón, C. N. de B. P. (5º 2010. (2010). Koha, metabuscadores y herramientas colaborativas de edición de contenidos. Recuperado a partir de <http://travesia.mcu.es/portalnb/jspui/handle/10421/4909>

Catálogo de la Biblioteca Nacional de España. (s. f.). Recuperado 16 de febrero de 2016, a partir de <http://catalogo.bne.es/uhtbin/cgiirsi/x/x/0/123>

CEAACES. (2015). Adaptación del modelo de evaluación institucional de universidades y escuelas politécnicas 2013 al proceso de evaluación acreditación y recategorización de universidades y escuelas politécnicas 2015. recuperado a partir de <http://www.ceaaces.gob.ec/sitio/wp-content/uploads/2013/10/adaptación-del-modelo-de-evaluación-institucional-de-univ.-y-esc.-politéc.-2013-al-proceso-de-eval-acredit-y-recateg-de-univers.-y-esc.-polit-201-5plenofinal-notif.pdf>

Chalon, P. X., Alexandre-Joaquim, L., Naget, C., & Becquart, C. (2006). Open your mind! Selecting and implementing an integrated library system: the open-source opportunity. Recuperado a partir de <http://eprints.rclis.org/12135/>

Charlton, G. (s. f.). Extending Koha using Linked Data. Recuperado 22 de enero de 2016, a partir de <http://zadi.librarypolice.com/~gmc/koha-linked-data/#/>

Cita Peñaranda. (s. f.). Conversaciones sobre biblioteca y web semántica - YouTube. Recuperado 27 de noviembre de 2014, a partir de <https://www.youtube.com/watch?v=P2-ijqZ5jRs>

ComputerWeekly. (2013, mayo 13). International Space Station adopts Debian Linux, drops Windows & Red Hat into airlock - Inspect-. Recuperado 19 de mayo de 2016, a partir de <http://www.computerweekly.com/blog/Open-Source-Insider/International-Space-Station-adopts-Debian-Linux-drops-Windows-Red-Hat-into-airlock>

Crum, C. (2010, abril 16). Choosing Open Source Solutions | Linux Journal [Revista]. Recuperado 7 de abril de 2016, a partir de <http://www.linuxjournal.com/content/choosing-open-source-solutions>

Cruz, D., Wieland, T., & Ziegler, A. (2006). Evaluation of Free/Open Source Software Products through Project Analysis - SPI_Paper_1_2.pdf. Recuperado 2 de marzo de 2015, a partir de https://intranet.hs-coburg.de/fileadmin/FB_EINF/staff/wieland/papers/SPI_Paper_1_2.pdf

Davis, H. C., Carr, L., Hey, J. M. N., Howard, Y., Millard, D., Morris, D., & White, S. (2010). Bootstrapping a Culture of Sharing to Facilitate Open Educational Resources. *IEEE Transactions on Learning Technologies*, 3(2), 96-109. <http://doi.org/10.1109/TLT.2009.34>

digibib/marc2rdf. (s. f.). Recuperado 16 de mayo de 2014, a partir de <https://github.com/digibib/marc2rdf>

Economic Impact of Public Libraries. (s. f.). Recuperado 19 de noviembre de 2013, a partir de http://pld.dpi.wi.gov/pld_econimpact

Egunjobi, R. A., & Awoyemi, R. A. (2012). Library automation with Koha. *Library Hi Tech News*, 29(3), 12-15. <http://doi.org/10.1108/07419051211241868>

El profesional de la informacion. (s. f.). Recuperado 19 de noviembre de 2013, a partir de http://www.elprofesionaldelainformacion.com/contenidos/1999/julio/tesauros_tesauros_automaticos_tesauros_automaticos_online.html

Estado, G. de T. de C. C. de las B. de la A. G. del, Arroyo Fernández, D., & Sellés Carot, A. (2010). Informe de evaluación del sistema integrado de gestión de bibliotecas Koha para las Bibliotecas de la Administración General del Estado. Recuperado a partir de <http://travesia.mcu.es/portalnb/jspui/handle/10421/4711>

Eyler, P. (2003, febrero 1). Koha: a Gift to Libraries from New Zealand | Linux Journal. Recuperado 6 de abril de 2016, a partir de <http://www.linuxjournal.com/article/6350>

Free Software Foundation. (2015). Lista de licencias con comentarios - Proyecto GNU - Free Software Foundation. Retrieved December 11, 2016, from <https://www.gnu.org/licenses/license-list.html>

Georgina Araceli Torres Vargas. (2014). Software libre : miradas desde la bibliotecología y estudios de la información (1.^a ed.). México: UNAM, Instituto de Investigaciones Bibliotecológicas y de la Información. Recuperado a partir de <http://132.248.242.6/~publica/conmutarl.php?arch=1&idx=287>

Goh, D. H.-L., Chua, A., Khoo, D. A., Khoo, E. B.-H., Mak, E. B.-T., & Ng, M. W.-M. (2006). A checklist for evaluating open source digital library software. *Online Information Review*, 30(4), 360-379. <http://doi.org/10.1108/14684520610686283>

Gómez-Rodríguez, G.-A. (2010). Descripción de un sistema integral de gestión bibliotecaria libre : PMB. Recuperado a partir de <http://eprints.rclis.org/20258/>

Goth, G. (2005). Open source infrastructure solidifying quickly. *IEEE Distributed Systems Online*, 6(3), -. <http://doi.org/10.1109/MDSO.2005.15>

Hinze, A., Buchanan, G., Bainbridge, D., & Witten, I. H. (2008). Semantics in Greenstone. University of Waikato Research Commons. Recuperado a partir de <http://researchcommons.waikato.ac.nz/handle/10289/1811>

Hinze, A., Buchanan, G., Bainbridge, D., & Witten, I. H. (s. f.). Greenstone: a platform for semantic digital.

Internet archive. (2009, noviembre). Operating system Family share for 11/2009 | TOP500

Supercomputing Sites. Recuperado 19 de mayo de 2016, a partir de <http://web.archive.org/web/20111108190659/http://www.top500.org/stats/list/34/osfam>

Kirk, S. A. (2007). Handbook of Research on Open Source Software: Technological, Economic, and Social Perspectives: Technological, Economic, and Social Perspectives. Idea Group Inc (IGI).

Krishnamurthy, M. (2008). Open access, open source and digital libraries: A current trend in university libraries around the world. Program: electronic library & information systems, 42(1), 48-55. <http://doi.org/10.1108/00330330810851582>

Kruk, S. R., & McDaniel, B. (2008). Semantic Digital Libraries. Springer.

Lannoy, B. (2009). Manuel modèle de sélection pour logiciels open source - Manuel_modele_OSS.pdf. Recuperado 6 de marzo de 2015, a partir de http://inventarisoss.smals.be/fr/164-RCH/version/default/part/AttachmentData/data/Manuel_modele_OSS.pdf

LibreSource. (2014, enero 2). En Wikipedia, the free encyclopedia. Recuperado a partir de <http://en.wikipedia.org/w/index.php?title=LibreSource&oldid=544658051>

Linux, S. J. V.-N. for, 16, O. S. | D., & now, 2014-- 13:13 GMTGet the newsletter. (s. f.). The fall of GPL and the rise of permissive open-source licenses. Recuperado 13 de enero de 2015, a partir de <http://www.zdnet.com/article/the-fall-of-gpl-and-the-rise-of-permissive-open-source-licenses/>

MagnusEnger/semantikoha. (s. f.). Recuperado 16 de mayo de 2014, a partir de <https://github.com/MagnusEnger/semantikoha>

Mangalam, H. (2010, mayo 23). How to Evaluate Open Source Software. Recuperado 18 de noviembre de 2014, a partir de http://moo.nac.uci.edu/~hjm/How_to_Evaluate_OSS.html

Mêgnigbêto, E. (2012). AUNIMARC bibliographic format database for ABCD. Program: electronic library and information systems, 46(4), 458-467. <http://doi.org/10.1108/00330331211276503>

Mendives, A., & Lizeth, K. (2006). Modelo de gestión estratégica para transformar las bibliotecas universitarias públicas de Lima Metropolitana en centros de gestión de la información. Universidad Nacional Mayor de San Marcos. Programa Cybertesis PERÚ. Recuperado a partir de <http://cybertesis.unmsm.edu.pe/handle/cybertesis/1747>

Merino, R. M. (2012, marzo 29). Hacia la Evaluación y Acreditación de Bibliotecas Universitarias: De la gestión de calidad de una biblioteca. Recuperado a partir de <http://haciaacreditaciondebibliotecasuniv.blogspot.com/2012/03/de-la-gestion-de-calidad-de-una.html>

Microsoft at the power point. (2003, septiembre). The Economist. Recuperado a partir de <http://www.economist.com/node/2054746>

Moreiro González, J. A., Rodríguez-Barquin, B. A., Martul, D. G., & Pinto, A. L. (2006). BIBLIOTECAS DIGITALES Y OPEN SOURCE SOFTWARE. Informação & Sociedade: Estudos, 16(1). Recuperado a partir de <http://www.ies.ufpb.br/ojs/index.php/ies/article/view/438>

Müller, T. (2011). How to choose a free and open source integrated library system. OCLC Systems & Services, 27(1), 57-78. <http://doi.org/10.1108/10650751111106573>

Muñoz, C. C., Velthuis, M. G. P., & Rubia, M. Á. M. de la. (2010). Calidad del producto y proceso software. Editorial Ra-Ma.

Openbrr.org. (2005). Business Readiness Rating for Open Source. Recuperado 2 de marzo de 2015, a partir de <http://docencia.etsit.urjc.es/moodle/mod/resource/view.php?id=4343>

Open Source Guide. (2016, abril 4). PMB / Libraries & Documentation. Recuperado 8 de abril

de 2016, a partir de <http://www.open-source-guide.com/en/Solutions/Applications/Libraries-documentation/Pmb>

Open source realigns priorities for «work» | Opensource.com. (s. f.). Recuperado 13 de febrero de 2016, a partir de https://opensource.com/open-organization/16/2/open-source-demonstrates-future-work?sc_cid=701600000011jJaAAI

Pacheco Arias, L. (2010). Arquitectura bibliotecaria entre la arquitectura e investigación (Thesis). Recuperado a partir de <http://dspace.utpl.edu.ec//jspui/handle/123456789/1151>

Pérez, T. S., & Lorca, J. G. (2003). Dentro de los portales bibliotecarios: flujo de trabajo en la automatización de bibliotecas. Scire: Representación y organización del conocimiento, 9(2), 37-60.

Piketty, T. (2014). El Capital En El Siglo XXI (Economía) (2014.^a ed.). Madrid: Fondo de Cultura Económica.

Pinedo Marín, L. E., & Medina Bonilla, L. E. (2010). Implementación de un sistema de integración para las bibliotecas municipales de Lima y Callao utilizando SOA y J2ME. Universidad Nacional Mayor de San Marcos. Programa Cybertesis PERÚ. Recuperado a partir de <http://cybertesis.unmsm.edu.pe/handle/cybertesis/2647>

PMB : programari lliure d'automatització de biblioteques. (2006, diciembre). [Text.Article]. Recuperado 19 de noviembre de 2013, a partir de <http://bid.ub.edu/17russo.htm>

Qualipso. (2010). Qualipso Open Source Maturity Model. Recuperado 13 de febrero de 2015, a partir de <http://sourceforge.net/projects/qualipso-omm/>

Rafiq, M., & Ameen, K. (2009). Issues and lessons learned in open source software adoption in Pakistani libraries. Electronic Library, The, 27(4), 601-610. <http://doi>.

org/10.1108/02640470910979561

Rocca Varela, A. B. (2011, febrero 26). Sistema Integrado de Gestión Automatizada de Bibliotecas basado en soluciones de software de Open Access para la Biblioteca del Departamento de Física, Facultad de Ciencias Exactas, Universidad Nacional de La Plata: Una actualización en la evolución de su aplicación. [Preprint]. Recuperado 26 de noviembre de 2013, a partir de <http://eprints.rclis.org/15403/>

Rocca Varela Adriana Beatriz Koha-BDF.pdf. (s. f.). Recuperado a partir de <http://eprints.rclis.org/15403/1/Rocca%20Varela%20Adriana%20Beatriz%20Koha-BDF.pdf>

Rodríguez-Gairín, J.-M., Sulé-Duesa, A., & Russo-Gallo, P. (2008). A virtual exhibition of open source software for libraries. Recuperado a partir de <http://eprints.rclis.org/11151/>

SCaLE 14x interview with Daniel Nazer, EFF | Opensource.com. (s. f.). Recuperado 18 de enero de 2016, a partir de https://opensource.com/business/16/1/scale14x-interview-daniel-nazer-eff?sc_cid=701600000011jJaAAI

Senso, J. (2011). Automatización de bibliotecas con PMB. Recuperado 8 de abril de 2016, a partir de <http://eprints.rclis.org/15335/1/pmb.pdf>

Smet, E. de, & Dhamdhere, S. N. (2010). ABCD: An Open Source Automation Tool for Libraries. *Pearl: A Journal of Library and Information Science*, 4(4), 215-219.

Stallman, R. (2004). *Software libre para una sociedad libre*. Madrid : Traficantes de Sueños, 2004. Recuperado a partir de <http://libros.metabiblioteca.org/handle/001/144>

Suárez-Potts, L. (2013). Open Source Library Systems in Europe: A Study | Joinup. Recuperado 3 de marzo de 2015, a partir de <https://joinup.ec.europa.eu/elibrary/case/open-source-library-systems-europe-study>

Valentín Ruiz, F. J., Fernández Sánchez, A. B., Torre-Marín, R., & Burgos, C. N. de B. P. (6º 2012. (2012). Servicios cloud computing para la gestión de bibliotecas: hacia un nuevo modelo. Recuperado a partir de <http://travesia.mcu.es/portalnb/jspui/handle/10421/6743>

Van den Berg, K. (2005). Finding Open options; An Open Source software evaluation model with a case study on Course Management Systems - Thesis.pdf. Recuperado 18 de febrero de 2015, a partir de <http://www.karinvandenberg.nl/Thesis.pdf>

Vimal Kumar, V. (2007). Selection and Management of Open Source Software in Libraries (pp. 1-5). INFLIBNET. Recuperado a partir de <http://eprints.rclis.org/8967/>

Webster, P. (2006). Interconnected and Innovative Libraries: Factors Tying Libraries More Closely Together. *Library Trends*, 54(3), 382-393. <http://doi.org/10.1353/lib.2006.0030>

What's behind Europe's love affair with open-source? | Network World. (s. f.). Recuperado 2 de diciembre de 2015, a partir de <http://www.networkworld.com/article/3008522/opensource-subnet/what-s-behind-europe-s-love-affair-with-open-source.html>

Wheeler, D. (2011, agosto 5). How to Evaluate Open Source Software / Free Software (OSS/FS) Programs. Recuperado 18 de noviembre de 2014, a partir de http://www.dwheeler.com/oss_fs_eval.html

Wheeler, D. (2014, mayo 8). Why Open Source Software / Free Software (OSS/FS, FOSS, or FLOSS)? Look at the Numbers! Recuperado 18 de noviembre de 2014, a partir de http://www.dwheeler.com/oss_fs_why.html

Witten, I. H., Bainbridge, D., & Boddie, S. J. (2001a). Greenstone: Open-Source Digital Library Software. *D-Lib Magazine*, 7(10). <http://doi.org/10.1045/october2001-witten>

Witten, I. H., Bainbridge, D., & Boddie, S. J. (2001b). Greenstone: open-source digital library

software with end-user collection building. *Online Information Review*, 25(5), 288-298. <http://doi.org/10.1108/14684520110410490>

Witten, I. H., Boddie, S. J., Bainbridge, D., & McNab, R. J. (2000). Greenstone: A Comprehensive Open-source Digital Library Software System. En *Proceedings of the Fifth ACM Conference on Digital Libraries* (pp. 113–121). New York, NY, USA: ACM. <http://doi.org/10.1145/336597.336650>

Zurita Sánchez, J. M. Z. (2011). El potencial del software libre en la gestión de información en bibliotecas. Em *Questão*, 16(1). Recuperado a partir de <http://200.144.189.42/ojs/index.php/revistaemquestao/article/view/7591>

ANEXOS

ANEXO A

GLOSARIO TERMINOLÓGICO

AACR : Anglo American Catalogation Rules, Reglas de Catalogación AngloAmericanas sirven para descripción bibliográfica y su intercambio.

CMMI: Capability maturity model integration o Modelo de integración de madurez de capacidades para mejora y evaluación de sistemas de software.

CODABAR: Es un formato de código de barras adaptado para ser leído ópticamente incluso si se ha impreso en impresoras de matriz de puntos, por lo que es bastante utilizado en bibliotecas.

COTS :Commercial off the Shelf. Se refiere a un producto estándar comprado y que se elabora en grandes cantidades para un mercado comercial. En el enfoque de TI es un producto de software comercial enfocado al público en general y de código propietario.

EDI : Electronic Data Interchange es un método de comunicación que proporciona estándares para intercambio de información electrónicamente.

FLOSS: Free Libre Open Source Software, término utilizado para describir a Software Libre de Código Abierto.

FRBR: Modelo conceptual entidad relación para el universo bibliográfico desarrollado por IFLA enfocado en generar un registro bibliográfico pensado en las necesidades de los usuarios.

FSF (Free Software Foundation): Fundación de Software Libre creada por Richard Stallman en 1985.

Gobernanza Abierta : Concepto que sostiene que más allá de solo el gobierno es el protagonista en la política, la gobernanza busca relaciones entre gobierno, instituciones y sociedad para que se implemente el derecho a conocer y participar en la toma de decisiones, con una arquitectura institucional y políticas transparente, rendición de cuentas y participación. Todos estos conceptos se apoyan en gran parte en la inversión en herramientas TIC para su consecución.

IFLA International Federation of Library Associations and Institutions : Federación Internacional de Asociaciones de Bibliotecas e Instituciones. Es una organización internacional que busca representar los intereses

de las bibliotecas, servicios de documentación, sus usuarios y los profesionales que se desempeñan en estas instituciones.

ILS Integrated Library System: Sistema Integrado de Gestión Bibliotecaria.

ISBD International Standard Bibliographic Description : Formato creado por la IFLA para la descripción bibliográfica y catalogación.

ISBN International Standard Book Number: Número Estándar Internacional de Libro es un identificador único para libros.

ISSN International Standard Serial Number: Número Internacional Normalizado de Publicaciones Periódicas es un identificador único para publicaciones seriadas o periódicas.

Kernel: Núcleo, en informática es la parte fundamental de un sistema operativo.

LDAP Lightweight Directory Access Protocol: Protocolo Ligero/Simplificado de Acceso a Directorios de formato abierto e independiente para acceder y mantener servicios de directorios distribuidos sobre protocolo IP. Generalmente se utiliza para almacenar usuarios y claves de acceso de una institución.

Licencia CECILL: Licencia derivada y compatible con GNU aplicada a la legislación francesa.

Licencia pública GNU: Es un tipo de licencia que declara como software libre y que protege a sus usuarios para que puedan disfrutar de las libertades especificadas en la misma.

MARC21 Abreviado : Es una subespecificación del formato MARC21 que no posee indicadores y subcampos.

MARC21: Machine Readable Cataloging, Catalogación Legible por Máquina es un formato creado por la Biblioteca del Congreso de Estados Unidos como un protocolo de descripción de campos bibliográficos adaptada para su uso a partir de el uso de computadores en el manejo de bibliotecas. Actualmente es el estándar ISO 2709.

MFHD MARC 21 Format for Holdings Data : Es una descripción detallada de cada tipo de dato, con ejemplos y convenciones de uso para ayuda en los procesos técnicos de catalogación bibliográfica.

OpenSearch : Es un conjunto de formatos para la compartición de resultados de motores de búsqueda tanto como clientes como servidores.

OpenURL : Es un estándar para la descripción de un recurso mediante un URL que sea accesible a los usuarios de Internet encontrar y acceder al documento y que es usado por los programas de gestión de bibliotecas para enlazarlos a sus catálogos.

RDA Resource Description and Access: Recursos de Descripción y Acceso es un nuevo código de catalogación que ha sustituido a las AACR2 con un enfoque en el mundo digital.

RDF Resource Description Framework: Marco de Descripción de Recursos es un modelo estándar para el intercambio de datos en la Web para modelado de información en la Web Semántica.

RSS Really Simple Syndication: Es un formato para compartir información actualizada en la Web con metadatos y basada en XML.

schema.org : Es una comunidad que busca crear, mantener y promover el uso de esquemas para datos estructurados en Internet.

SIP2 Standard Interchange Protocol: Es un estándar propietario de comunicación entre software para bibliotecas desarrollado por 3M.

unAPI : Es una pequeña API para webapps HTML para copublicar objetos de forma estandarizada.

Unicode : Estándar de codificación de caracteres para cualquier lengua.

WCAG Web Content Accessibility Guidelines: Son normas desarrolladas por el consorcio W3 y otras instituciones que buscan cumplir con estándares de accesibilidad para personas, organizaciones y gobiernos en todo el mundo, en especial de las personas con discapacidad.

Z39.50 : Protocolo cliente servidor a nivel de capa de aplicación para buscar y recuperar información de una base de datos a través de TCP/IP y es utilizado por sistemas de gestión bibliotecaria para intercambio de información bibliográfica.

ANEXO B

ENCUESTA DE BIBLIOTECAS UNIVERSITARIAS

Se contactó con varias bibliotecas universitarias y finalmente se pudo realizar la investigación con las siguientes:

- Universidad Metropolitana
- Pontificia Universidad Católica Sede Quito
- Pontificia Universidad Católica Sede Ambato
- Universidad Técnica de Cotopaxi
- Universidad Central del Ecuador, facultad de Psicología
- Universidad Técnica de Ambato, facultad de Auditoría
- Universidad Técnica de Ambato, facultad de Ciencias Administrativas
- Universidad Andina Simón Bolívar
- Universidad San Francisco de Quito
- FLACSO Facultad Latinoamericana de Ciencias Sociales
- ESPE Escuela Politécnica del Ejército Sede Latacunga
- ESPE Escuela Politécnica del Ejército Sede Quito
- Universidad del Pacífico
- UDLA Universidad de las Américas

Las preguntas planteadas fueron 5:

1. Qué programa de gestión de bibliotecas utilizan y desde hace cuanto tiempo?
2. Que cantidad de libros o artículos tienen clasificado dentro de ese sistema?
3. Realizan catalogación de materiales físicos exclusivamente o también digitales?
4. De su experiencia con los usuarios, cual es la percepción de ellos sobre el sistema, lo usan, pueden hacer consultas a través de Internet?
5. En su experiencia personal cuáles son las principales ventajas y desventajas que tiene el sistema que utilizan?

Recolección y descripción de la información

Con la tabulación de los datos de las preguntas realizadas se transcriben las respuestas escritas en las preguntas abiertas para poder procesarlas en registros y categorizarlas según el tema.

Categorización y análisis de contenidos Universidad

Universidad	Software	Tiempo de uso (años)	Libros / artículos clasificados	Materiales físicos / digitales
USFQ	ALEPH 500	9	109270	Ambos
UTC Cotopaxi	SIABUC 9	14	8703	Ambos
UTA Ambato Fac. CCAA	ABCD	1	3600	Físicos
UTA Ambato Fac. Auditoria	Library	5	10000	Físicos
Upacífico	Siabuc	6	10754	Físicos
U. Metropolitana	KOBLI	1	8685	Físicos
UDLA	Symphony	11	25000	Físicos
UC Fac. Psicología	WinISIS	3	1304	Ambos
UASB	Bibliosib / WinISIS	5	155000	Ambos

ESPE Latacunga	SIABUC 9	14	15000	Ambos
FLACSO Sede Ecuador	Bibliotec – Basado en Open Biblio	5	7500	Ambos
PUCE Quito	Propio	3	95000	Ambos
PUCE Ambato	SIABUC 9	3	10000	Ambos
ESPE Quito	Siabuc 9	11	40000	Ambos
Promedio		6,5	35.701	Ambos: 9 Físico: 5
Moda		3	10000	
Mediana		5	10377	

Software	Open Source si-no	Cantidad de bibliotecas
SIABUC	No	5
Aleph	No	1
Bibliotec	Si	1
KOBLI	Si	1
Propio	No	1
Simphony	No	1
WinIsis	Si	1
Bibliosib	Si	1
ABCD	Si	1
Library	No	1
TOTAL	Si: 5 No: 9	14

Resumen encuestas software de bibliotecas universitarias.

Universidad	Software	Tiempo de uso (años)	Libros / artículos clasificados	Materiales físicos / digitales	Percepción de los usuarios	Principales ventajas	Principales desventajas
USFQ	ALEPH 500	9	109270	Ambos	Nunca ha habido quejas de los usuarios por el manejo del software	<p>Flexible y confiable se adapta a las necesidades de la biblioteca.</p> <p>Fácil de usar flujos de trabajo amigables e interfaces gráficas intuitivas.</p> <p>Permite la personalización de cualquier requerimiento en la biblioteca.</p> <p>Conectividad e interacción transparente con otros sistemas y bases de datos.</p>	No he encontrado ninguna desventaja por ahora
UTC Cotopaxi	SIABUC 9	14	8703	Ambos	<p>Es un sistema amigable que puede ser reconocido y utilizado sin mayor problema por los usuarios aún por los que no tienen experiencia.</p> <p>Si permite hacer consultas y reservaciones a través de internet.</p>	<p>Está dirigido a hispanohablantes</p> <p>Es versátil. Cada uno de los módulos contiene campos que abarcan casi todo el proceso bibliotecológico de los recursos de información.</p> <p>Es fácil de ser utilizado por la sencillez de cada uno de sus módulos. Pueden hacer uso de éste, los jóvenes, niños y adultos mayores.</p> <p>Se adapta a casi todas las necesidades de las bibliotecas de Latinoamérica</p> <p>Los problemas están siendo monitoreados frecuentemente y enmendados por los responsables del</p>	<p>No es un software de libre difusión y empleo, por lo tanto, a mediano plazo en las instituciones públicas del país no podrá ser utilizado.</p> <p>No se permite exportar registros más que de una versión a otra del mismo software</p> <p>La migración de datos se produce con un margen de 20-30% de error</p> <p>Debe permitir la configuración de un módulo para adaptar a la necesidad de cada biblioteca, por ejemplo, para automatizar</p>

						software en la Universidad de Colima.	documentos de archivo, documentos fotográficos, piezas de museo, etc.
UTA Ambato Fac. CCAA	ABCD	1	3600	Físicos	Todavía no lo manejan muy bien	La principal desventaja, cuando no se tiene internet, se pierde todo Al migrar la información, prácticamente toca registrar todos los campos	Los prestamos automatizados Facilidad en búsquedas
UTA Ambato Fac. Auditoria	Library	5	10000	Físicos	Desde luego, los usuarios lo usan continuamente por su facilidad en el manejo, realizan búsquedas directamente desde la página web y cuando ingresen la información de los recursos electrónicos podrán visualizar y descargar directamente desde la web.	La catalogación se vuelve por demás sencilla por cuanto este sistema dispone de herramientas y materiales que como bibliotecarios manejamos, por ejemplo: contiene la Tabla Cutter's , el sistema de clasificación que la biblioteca este utilizando, en nuestro caso, o en las bibliotecas de la Universidad el CDU. Podemos sacar al instante diferentes tipos de estadísticas y reportes. Tiene control para el servicio de Internet. Control de visitas a la biblioteca. Cumple con los requerimientos exigidos por el CEAACES Disponemos del servicio técnico inmediato, cuando sea necesario	No es software libre. Es un sistema que no es de libre descarga
Upacífico	Siabuc	6	10754	Físicos	No lo hacemos a traves de internet, nos falta implementarlo, solo lo manejamos con acceso directo al modulo de consultas	es amigable y práctico	que el inventario es limitado, solo vota ciertos indicadores, titulo, autor, clasificación y por biblioteca
U. Metropolitana	KOBLI	1	8685	Físicos	El Catálogo esta subido a la página web de la Universidad y se lo puede consultar a través de internet es amigable con el usuario y por el momento se ha estado capacitando en el uso del	Las ventajas es que es amigable con el usuario, es gratuito, permite hacer un gran número de cambios lo que permite a la biblioteca adaptarlo a sus necesidades específicas.	La desventaja es que al ser gratuito las mejoras al sistema son lentas, no permite mantener el catálogo fuera del internet lo que ocasiona problema si no existe la conexión a la red o por algún motivo se va la

					mismo.		luz.
UDLA	Symphony	11	25000	Físicos	Al momento estamos en la fase de implementación, el sistema no está a disposición de nuestros usuarios. Estará disponible a partir de mediados de marzo.	Permite que los usuarios realicen búsquedas en un solo cajón de búsqueda, reservas en línea, creación de una biblioteca personalizada, bibliografías, etc.	Por el momento no hemos detectado desventajas
UC Fac. Psicología	WinISIS	3	1304	Ambos	Consulta permanente en los catálogos en línea, facilitando... búsquedas temáticas especializadas	Gratuito, Auspicio de la UNESCO, es amigable, fácil acceso a la información, búsqueda rápida, identifica todos los documentos de la temática consultada, busca por términos no importa el orden ni la ortografía. Bibliográficos, Elabora catálogos. Se puede diseñar diversas bases de datos.	No ha sido explorado lo suficiente para optimizarlo al máximo. No hay actualizaciones frecuentes.
UASB	Bibliob / WinISIS	5	155000	Ambos	Nos ha ido muy bien con el sistema, es amigable con el usuario. Permite buena recuperación.	Que es un sistema basado en software libre muy trabajado en la Región como es el ISIS y su familia, por tanto puede ser modificado según nuestras necesidades. Usa interfaces gráficas. Permite automatización de procesos de préstamo e inventarios, tiene interconectividad con el sistema académico de la universidad. Permite visibilidad en internet. Se puede enlazar a recursos electrónicos. Permite migración de formatos, así desde CEPAL a MARC, que es lo que hicimos.	Al ser una adaptación, dependemos de un técnico que las realice y aunque hay soporte permanente, eso nos limita como institución. Por otro lado creo que podríamos usar software propietario que ya tenga incorporado protocolos como el Z39-50 y control de autoridades, que este software no lo tiene. Se entiende que este software es de aplicación escalable y modular, pero solamente tenemos los módulos relacionados con la catalogación y préstamo faltándonos los otros módulos de gestión estadística, etc. eso es una limitación grande. Además, en mi opinión muy personal es un software un tanto inestable.
ESPE	SIABU	14	15000	Ambos	SIABUC 9 es un sistema muy	Atención eficaz a los usuarios	No hay soporte técnico personal

Latacunga	C 9				fácil de utilizar, especialmente el catálogo, los estudiantes no tienen ningún inconveniente en el uso dentro de la biblioteca o a través de internet desde el sitio en que se encuentren pueden hacer sus consultas, reservaciones de materiales e inclusive consultar el documento a texto completo.	Que es un sistema que maneja por procesos Catalogo en línea y eficaz recuperación de la información Reporte de estadísticas Formato MARC, protocolo z3950 Actualización gratuita	Que es un sistema cerrado no se puede modificar nada
FLACSO Sede Ecuador	Biblioteca – Basado en Open Biblio		7500	Ambos	La interfaz de búsqueda para el usuario es muy fácil de usar y las búsquedas a través de internet tienen una velocidad de respuesta bastante aceptable	El uso del sistema es muy fácil y adaptable a las necesidades de la biblioteca física y virtual	El software no se ha actualizado desde el 2011
PUCE Quito	Propio		95000	Ambos	Crean que es bueno y lo usan a través de Internet	Las de las TICs,.	Paquetes subutilizados, no aprovechados totalmente por varias razones de parte de bibliotecarios y usuarios
PUCE Ambato	SIABU C 9		10000	Ambos	Una vez implementado todos los servicios, es muy funcional para el usuario final ya que integra transacciones web muy útiles.	Bajo costo Fácil implantación. Muy intuitivo en su uso. Integral (Abarca todas las áreas de gestión bibliotecaria). Posee soporte en línea y manuales. Brinda actualizaciones. Uso de estándares de ágil transferencia de información. Facilidad en alteración de base de datos	Aún no es web. Falta de integración a tecnologías como RFID. Considerable cantidad de errores. Escasa seguridad. Inexistencia de reportes personalizados. Soporte muy ineficiente.
ESPE Quito	Siabuc 9	11	40000	Ambos	Si, hemos realizado educación a usuarios a fin de que los recursos sean utilizados por lo general el usuario desconoce que significan	Ventajas muchas es de fácil uso, amigable, permite realizar muchas opciones, como: la búsqueda, inventarios, estadísticas, control de	

				los números ubicados en los libros. Con la Educación a usuarios facilita la identificación del documento en las estanterías, además al brindar un servicio de estantería abierta esto agiliza encontrar los recursos requeridos.	préstamos, adquisiciones, entre otras	
--	--	--	--	--	---------------------------------------	--