



Universidad
Zaragoza

Trabajo Fin de Grado

Software libre para Sistemas Integrados de Gestión
Bibliotecaria: estado de la cuestión y propuestas

Free Software for Integrated Library Systems: state
of the art and proposals

Autor

David Navarro Ruiz

Director

Antonio-Paulo Ubieto Artur

FACULTAD DE FILOSOFÍA Y LETRAS

2019

Software libre para Sistemas Integrados de Gestión Bibliotecaria: estado de la cuestión y propuestas

David Navarro Ruiz

Vº Bº del Director

Firma del autor

Fdo.: Antonio-Paulo Ubieto Artur

Fdo.: David Navarro Ruiz

NAVARRO RUIZ, David

Software libre para Sistemas Integrados de Gestión Bibliotecaria: estado de la cuestión y propuestas [Manuscrito] / David Navarro Ruiz ; director Antonio-Paulo Ubieto Artur. – 2019. – 87 p.: il. ; 30 cm. – Trabajo Fin de Grado de Información y Documentación de la Universidad de Zaragoza, 2019.

1. Software libre. 2. Sistemas Integrados de Gestión Bibliotecaria. I. Ubieto Artur, Antonio-Paulo. II. Tít.

004.6:025



Registrado por la [Universidad de Zaragoza](#) bajo la licencia [Creative Commons](#)

Resumen

El uso de software libre en los procesos de automatización de las bibliotecas ha ido ganando protagonismo frente a las soluciones propietarias. Ante el creciente interés de profesionales, tanto en formación como consolidados, el presente trabajo ofrece una panorámica general de Sistemas Integrados de Gestión Bibliotecaria (SIGB) de software libre, principales características y funcionalidades, además de un conjunto de plataformas destinadas a la instalación de los mismos para posibilitar su evaluación, aprendizaje y posible implementación en un futuro.

Palabras clave

Sistemas Integrados de Gestión Bibliotecaria, Software libre, Software de código abierto
Automatización de bibliotecas

Abstract

The use of free software in library automation processes has gained prominence over proprietary solutions. Given the growing interest of professionals, both during training or already consolidated, this work offers a general overview of free software Integrated Library Systems (ILS), their main features and functionalities, as well as a set of platforms for their installation in order to enable their evaluation, learning and possible implementation in a future.

Keywords

Integrated Library System, Free software, Open source software, Library automation

Índice

1	Introducción.....	1
1.1	Justificación	1
1.2	Objetivos	3
1.3	Metodología	3
1.3.1	Definición de las necesidades informativas del trabajo.....	3
1.3.2	Búsqueda de SIGBs de software libre	4
1.3.3	Delimitar los SIGBs de software libre recuperados	4
1.3.4	Plataformas de instalación y prueba de los SIGBs	4
1.3.5	Estudio de los SIGBs de software libre recuperados.....	4
1.4	Marco teórico	7
1.4.1	Automatización de bibliotecas	7
1.4.2	Sistemas Integrados de Gestión Bibliotecaria (SIGB)	9
1.4.3	Software libre, gratuito, código abierto y propietario	11
2	PLATAFORMAS DE INSTALACIÓN Y PRUEBA DE LOS SIGBS	17
2.1	AMPdoc	17
2.2	XAMPP.....	19
2.3	LAMP	21
2.4	Virtualización.....	22
2.5	Otros medios	24
3	Análisis y discusión de resultados	26
3.1	ABCD	29
3.2	BIBLIO	31
3.3	Evergreen	33
3.4	Koha.....	36
3.5	Meran	42
3.6	OpenBiblio.....	44
3.7	PMB	47

4	Comparación de los SIGBs.....	51
5	Conclusiones.....	53
6	Bibliografía.....	57
7	Anexos	63
	Anexo A: Matriz técnica para la evaluación de software libre: generalidades	63
	Anexo B: Matriz técnica para la evaluación de software libre: funcionalidades	65
	Anexo C: Derechos y responsabilidades del consumidor	67
	Anexo D: Licencias de software libre existentes en la actualidad	69
	Anexo E: Instalaciones de Koha en España y Latinoamérica	75
	Anexo F: Instalaciones de Meran	77

Índice de Ilustraciones

Ilustración 1: Mapa de una biblioteca "libre".....	1
Ilustración 2: Pantalla principal de AMPdoc.....	17
Ilustración 3: SIGBs disponibles en AMPdoc	18
Ilustración 4: Panel de control de XAMPP	19
Ilustración 5: Apache y MySQL en funcionamiento.....	20
Ilustración 6: Intranet de Openbiblio.....	20
Ilustración 7: Intranet de PMB	21
Ilustración 8: OPAC de PMB	22
Ilustración 9: Intranet Koha-Kobli	23
Ilustración 10: OPAC de Koha-Kobli	24
Ilustración 11: Mapa de instalaciones de ABCD.....	29
Ilustración 12: Mapa de instalaciones de Evergreen	34
Ilustración 13: Mapa de instalaciones de Koha	38
Ilustración 14: Mapa de instalaciones de Koha en España.....	40
Ilustración 15: Mapa de instalación de Meran.....	42
Ilustración 16: Mapa de instalaciones de OpenBiblio	45
Ilustración 17: Mapa de instalaciones de PMB	48

Índice de Tablas

Tabla 1: Generalidades del SIGB	5
Tabla 2: Requisitos de instalación del SIGB	6
Tabla 3: Funcionalidades del SIGB.....	6
Tabla 4: Principales diferencias entre software libre y propietario	15
Tabla 5: Lista de SIGBs recuperados	29
Tabla 6: Generalidades de ABCD	30
Tabla 7: Requisitos de instalación de ABCD	31
Tabla 8: Funcionalidades de ABCD	31
Tabla 9: Generalidades de BIBLIO	32
Tabla 10: Requisitos de instalación de BIBLIO.....	33
Tabla 11: Funcionalidades de BIBLIO.....	33
Tabla 12: Generalidades de Evergreen	35
Tabla 13: Requisitos de instalación de Evergreen.....	35
Tabla 14: Funcionalidades de Evergreen.....	36
Tabla 15: Generalidades de Koha.....	41
Tabla 16: Requisitos de instalación de Koha.....	41
Tabla 17: Funcionalidades de Koha	41
Tabla 18: Generalidades de Meran	43
Tabla 19: Requisitos de instalación de Meran.....	43
Tabla 20: Funcionalidades de Meran	44
Tabla 21: Generalidades de OpenBiblio.....	46
Tabla 22: Requisitos de instalación de OpenBiblio.....	46
Tabla 23: Funcionalidades de OpenBiblio	47
Tabla 24: Generalidades de PMB	49
Tabla 25: Requisitos de instalación de PMB	49
Tabla 26: Funcionalidades de PMB	50
Tabla 27: Comparación de módulos de los SIGBs.....	51
Tabla 28: Comparación de requerimientos de los SIGBs.....	52
Tabla 29: Estándares para el intercambio de información bibliográfica	52
Tabla 30: Generalidades del SIGB	64
Tabla 31: Funcionalidades del SIGB	66
Tabla 32: Derechos y responsabilidades del consumidor.....	68
Tabla 33: Licencias de software libre.....	73
Tabla 34: Instalaciones de Koha.....	76

1 INTRODUCCIÓN

1.1 Justificación

Las bibliotecas son un centro neurálgico de acceso y difusión de todo tipo de información sin mayor restricción que pertenecer a su comunidad de usuarios. Desde hace unos años, éstas han decidido apostar por el modelo de *Open Access* o Acceso Abierto, el cual permite el acceso gratuito a la información y al uso sin restricciones de los recursos digitales.

Simultáneamente, el movimiento del software libre se ha ido imponiendo en los servicios a la comunidad, incluyendo bibliotecas y otras unidades de información, poniendo a disposición herramientas y aplicaciones gratuitas y de gran potencia basadas en proyectos colaborativos y que ofrecen enormes posibilidades a las unidades y, consecuentemente, a sus usuarios. Estas son usadas diariamente por las bibliotecas favoreciendo así la creación del modelo de biblioteca “libre” e incluye, entre otras, aplicaciones como *Greenstone*, para crear colecciones digitales; *Wordpress*, para elaborar blogs o *Alfresco* para administrar y gestionar documentos.

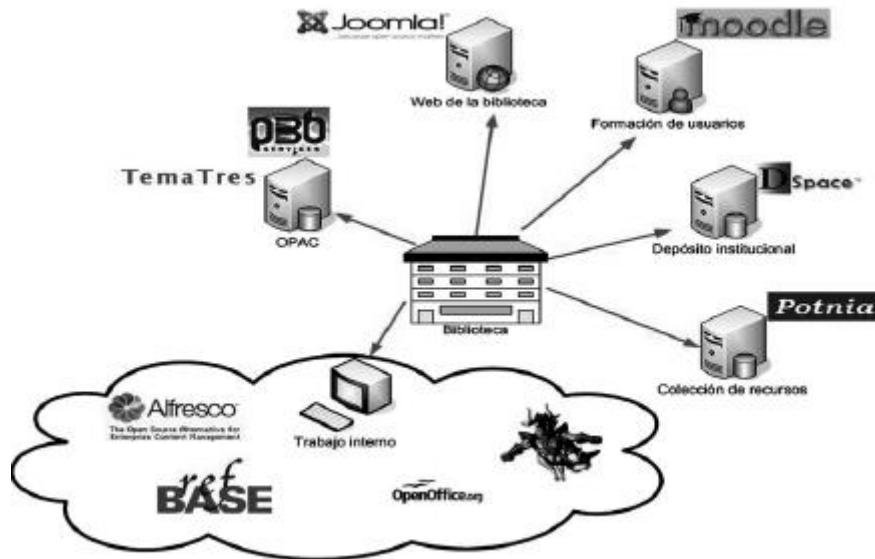


Ilustración 1: Mapa de una biblioteca "libre". Fuente: Patricia Russo, 2009

Dentro de estas aplicaciones encontramos los “Sistemas Integrados de Gestión Bibliotecaria” (en adelante, SIGB), los cuales automatizan de forma integrada todos los procesos y servicios básicos de la biblioteca en un único sistema que centraliza, optimiza y rentabiliza la introducción y gestión de los datos (González-Fernández, 2009). Hasta hace unos años, el mercado de los SIGBs estaba completamente dominado por los de tipo propietario. Sin embargo, actualmente cada vez más bibliotecas optan por SIGBs de software libre.

La economía ha sido un factor determinante a la hora de impulsar a las bibliotecas a elegir estos SIGBs frente a propietarios. Müller (2008), explica lo difícil que puede resultar para muchas bibliotecas adquirir un SIGB propietario, y es que su coste medio fluctúa entre los 18.000 y 65.000 € y entre 1.300 y 6.500 € únicamente la licencia anual. A su vez, Breeding (2017) muestra un enfoque similar al de Müller, recalca la inviabilidad en cuanto al mantenimiento de los SIGBs propietarios por parte de las bibliotecas debido a los cada vez más escasos presupuestos de los que éstas disponen, y explica que estas dificultades económicas son un importante factor para el desarrollo de nuevos sistemas integrados de software libre.

Otra de las principales razones es que contratar un producto propietario conlleva, además de pagar la adquisición y las licencias de uso, quedar ligado con la empresa propietaria; si el servicio no es satisfactorio quedan como opciones continuar con la empresa o migrar a otro SIGB, con las dificultades que esto pueda suponer. En cuanto al software libre, si el servicio no resultara satisfactorio y ha sido contratado con una empresa concreta, la ruptura del contrato con esa empresa no supone el abandono del producto, ya que es posible la contratación del servicio técnico con otra empresa distinta.

A pesar de lo nombrado anteriormente y aunque los SIGBs de código abierto han ganado una gran popularidad en los últimos años, estudios como el de Cháfer (2015), centrado en la situación española, o informes como el de Breeding (2017), muestran que la industria de la automatización de bibliotecas sigue dominada por los sistemas propietarios.

Uno de los principales motivos es el desconocimiento o la falta de comprensión acerca de la naturaleza del software libre. La comunidad bibliotecaria se ha acostumbrado a trabajar con proveedores comerciales y es muy difícil que se incorporen a esta tendencia tan reciente. Es complicado que los bibliotecarios comprendan que al utilizar software propietario se comportan como simples consumidores, mientras que con estas aplicaciones de software libre pueden ser partícipes del desarrollo del programa y mejorarlo en beneficio de la comunidad.

Además, el bibliotecario que quiere implantar un SIGB de software libre se encuentra ante un mercado en constante evolución y transformación. Una de las tareas más complicadas a los que debe enfrentarse es evaluar y seleccionar cuál es la mejor opción entre las distintas alternativas de aplicaciones o paquetes de software para establecer cuál es el idóneo para la biblioteca. Moreiro (2011) afirma que ninguna aplicación va a responder al completo a todas las necesidades de una organización específica; sin embargo, seleccionar el software libre y de código abierto permite adaptarlo, actualizarlo e integrarlo a los posibles cambios que puede sufrir ésta.

Todo esto justifica la pertinencia de este trabajo, el cual pretende aportar una panorámica de los SIGBs de software libre disponibles en el mercado y, de esta forma, servir como un punto de partida para aquellos bibliotecarios, personas o entidades que estén interesadas en implantar uno en su unidad de información.

Asimismo, esperamos que el presente trabajo pueda ser de utilidad como apoyo a la docencia de estas materias en las distintas enseñanzas en Grados en Información y Documentación, como el de la Universidad de Zaragoza.

1.2 Objetivos

a) Objetivo general

El principal objetivo del presente trabajo es ofrecer una panorámica de los SIGBS de software libre disponible en la actualidad, y algunas de sus características y funcionalidades.

b) Objetivos específicos

Examinar las diferentes plataformas de instalación y prueba para los SIGBs y así servir de orientación, tanto en entornos de enseñanza como profesionales.

Realizar una matriz comparativa de las aplicaciones, tanto de sus funcionalidades como de sus requisitos de instalación, entre otros parámetros.

1.3 Metodología

Con el fin de cumplir los objetivos nombrados anteriormente, la metodología del trabajo se divide en varias etapas:

1.3.1 Definición de las necesidades informativas del trabajo

Obtención de información sobre la automatización de bibliotecas, los SIGBs y los diferentes tipos de software que pueden conformar éstos. Para ello se han consultado las siguientes fuentes para seleccionar la bibliografía:

- El catálogo ROBLE de la Biblioteca de la Universidad de Zaragoza
- El catálogo de recursos electrónicos de ROBLE, desde donde se ha accedido a las bases de datos Dialnet y Scopus
- Recursos de acceso abierto como E-LIS y Google Académico
- Sitios webs especializados como FOSS4Lib¹, dedicado íntegramente al software libre y de código abierto para bibliotecas
- Recursos web 2.0 como el grupo de “Software libre para biblioteca²” en Facebook o el blog personal de Jesús Tramullas “Tramullas.com³”, el cual contiene varias entradas dedicadas al software libre y, el de Alonso Arévalo “Universo abierto⁴”, con una categoría específica dirigida al software libre.

Tras la consulta de estas fuentes se obtuvieron una serie de resultados, donde destacan la gran cantidad de artículos de revistas científicas y los estudios de casos donde se describen con gran detalle las experiencias en cuanto a la implantación de un SIGB concreto, criterios de selección de estos o migraciones desde sistemas propietarios.

También debemos destacar la obra de García Melero y García Camarero (1999), así como el manual de referencia de Dania Bilal (2014) y por últimos los numerosos trabajos de

¹ <https://foss4lib.org>

² <https://www.facebook.com/groups/softwarefree/>

³ <https://tramullas.com/>

⁴ <https://universoabierto.org/>

Óscar Arriola Navarrete sobre el software libre, que han sido de gran ayuda para el desarrollo de este trabajo.

1.3.2 Búsqueda de SIGBs de software libre

En primer lugar, se ha usado como referencia la obra de Dania Bilal (2014) en la cual se numeran varios, en segundo lugar, FOSS4Lib el cual posee una sección dedicada exclusivamente a los SIGBs de software libre, el grupo de Facebook y los dos blogs anteriormente nombrados y, en última instancia, los buscadores Google, Ask, Bing y Yahoo!

1.3.3 Delimitar los SIGBs de software libre recuperados

Se definieron tres criterios para delimitar el software a ser analizado con mayor profundidad en este trabajo: a) software en español o multilingüe con opción de español, b) última actualización no superior a cinco años de antigüedad y c) software libre y disponible para descarga y uso.

1.3.4 Plataformas de instalación y prueba de los SIGBs

Se estudiaron y/o instalaron algunos de los SIGBs de software libre recuperados utilizando diferentes plataformas: AMPdoc, XAMPP (ambas WAMP), LAMP, virtualización y otras.

1.3.5 Estudio de los SIGBs de software libre recuperados

Para el estudio de las aplicaciones se definió una matriz con una serie de parámetros, los cuales fueron distribuidos en tres grupos:

- El primero de ellos contiene nueve (9) parámetros referentes a las características generales que identifican la aplicación como, por ejemplo, el nombre de la aplicación, el país de origen, etc.
- El segundo grupo contempla cuatro (4) parámetros que se refieren a los requisitos de instalación, es decir, a los componentes que debe disponer nuestro equipo para un funcionamiento adecuado de la aplicación.
- Por último, se incluyen en el último grupo ocho (8) parámetros que determinan la funcionalidad de la aplicación a partir de los módulos que esta posee o no.

Para la elaboración de las tablas de “generalidades” y “funcionalidades” se tomaron como referencia las tablas de la matriz técnica para la evaluación de software libre de Fernández Morales y Chinchilla Arley (2013) (véanse anexos A y B). La tabla de “requisitos de instalación” es de elaboración propia.

a) Tabla de generalidades:

Característica	Descripción
Nombre de la aplicación	Nombre normalizado de la aplicación.
Versión	Versión de la aplicación. Una misma aplicación puede estar funcionando en diferentes organizaciones, pero con versiones diferentes, lo cual las convierte en aplicaciones distintas. Se seleccionará la versión más reciente.
País de origen	País donde la aplicación fue desarrollada originalmente.
Idiomas soportados	Si se trata de una aplicación multilingüe o de solo un idioma.
Creador	Creador o desarrolladores de la aplicación.
Licencia	Se refiere al tipo de licencia de la aplicación para determinar si se trata de software libre o no y conocer que usos podemos hacer con la aplicación. Según Müller (2011) siempre debemos elegir software libre de código abierto.
Página web	Dirección URL de la página web de la aplicación.
Página de descarga	Dirección URL de su página de descarga.
Soporte	Si existe disponibilidad o no de soporte: documentación técnica, tutoriales, foros, listas de correo, etc. Según Müller (2011) se debe tener en cuenta la comunidad de usuarios que hay detrás de una aplicación, ya que de dicha comunidad va a depender el apoyo que tengamos en la instalación y uso de la aplicación, como el desarrollo futuro de esta.

Tabla 1: Generalidades del SIGB

b) Tabla de requisitos de instalación:

Requisito	Descripción
Sistema Operativo	Se refiere a los sistemas operativos en los que se puede ejecutar la aplicación.
Servidor	La aplicación en la que se instala el software del sistema de gestión bibliotecario y la que almacena, gestiona y proporciona control de acceso para los objetos almacenados.
Lenguaje de programación	Corresponde a los lenguajes de programación utilizados para su desarrollo

Requisito	Descripción
Sistema Gestor de Base de Datos	El sistema que permite la creación, gestión y administración de bases de datos.

Tabla 2: Requisitos de instalación del SIGB

c) Tabla de funcionalidades:

Funcionalidad	Descripción
Módulo de administración	El módulo de administración es aquel que se encarga de todas las funciones administrativas, de configuración y parametrización de la aplicación.
Módulo de adquisiciones	El módulo responsable de facilitar y registrar los procesos de intercambios, compras y donaciones en una unidad de información.
Módulo de catalogación	El módulo que incluye las funciones relativas a los procesos técnicos relacionados con el desarrollo y mantenimiento de los catálogos bibliográficos.
Módulo de circulación	El módulo responsable de todos los procesos relativos al préstamo y devoluciones de materiales por parte de los usuarios.
Módulo de estadísticas	El módulo que permite conocer estadísticamente las operaciones realizadas en la unidad de información. Operaciones como préstamos, informes de reservas activas, listados de libros dañados, usuarios con mayor actividad y el uso de los distintos servicios ofrecidos en la unidad de información.
Módulo de OPAC	El Catalogo Público en línea u OPACS es un catálogo automatizado de acceso público en línea de los materiales de una biblioteca desde sitios remotos a través de la Web.
Módulo de publicaciones periódicas	El modulo que se utiliza para hacer el seguimiento de las revistas, los periódicos y otros ítems recibidos de acuerdo a un calendario regular.
Otros módulos	Se incluirán módulos poco comunes o específicos de un SIGB.

Tabla 3: Funcionalidades del SIGB

Finalmente se utilizaron los mapas de instalaciones Library Technology⁵ para observar donde se encuentran instalados los SIGBs analizados. Cabe destacar que no es una herramienta exacta, pero se utilizó de forma orientativa.

⁵ <https://librarytechnology.org/libraries/map.pl>

1.4 Marco teórico

Con el propósito de aportar una fundamentación teórica, se desarrollan los temas de “Automatización de bibliotecas”, “Sistemas Integrados de Gestión Bibliotecaria” y, por último, los distintos tipos de software que pueden conformar estos.

1.4.1 Automatización de bibliotecas

1.4.1.1 Concepto de automatización

La Real Academia Española define el término “automatización” como “acción y efecto de automatizar”, y éste a su vez, define “automatizar” como “la aplicación de la automática a un proceso o a un dispositivo”. Ambas definiciones resultan incompletas y no aportan la claridad necesaria para poder comprender el concepto en su totalidad. En párrafos siguientes se muestran una serie de definiciones más detalladas sobre el concepto de automatización.

Buonocore (1976), en su Diccionario de Bibliotecología, define el concepto de automatización como “el proceso que tiende hacia la mecanización de las actividades industriales, de las comunicaciones, etc. Esta automatización creciente desempeña una función importante en la técnica documentalista, gracias al empleo de máquinas electrónicas de calcular y de otras llamadas elaboradoras de datos”.

Para López Yepes (2004) la automatización es “la tecnología que trata de la aplicación de sistemas mecánicos, electrónicos y de bases computacionales para operar y controlar la producción”.

Lorenzo (2006) entiende la automatización como “un conjunto de técnicas asociadas con la aplicación de sistemas de tipo mecánico/electrónico y basado en ordenadores, cuyo objetivo es la operación y control de la producción”.

Con estas definiciones se puede concluir que la automatización es un elemento esencial en los procesos de producción y que facilita, mejora y simplifica tareas repetitivas y/o cotidianas que realiza el hombre gracias al uso de computadoras, máquinas y herramientas informáticas.

El concepto de automatización surge en el siglo XVIII con la Revolución Industrial, momento en el cuál cambió radicalmente la forma de trabajar y comienza la automatización industrial. Pero no fue hasta el siglo XX cuando la automatización traspasa las barreras de la industria y pasa a formar parte de lo cotidiano, logrando facilitar significativamente muchas de las tareas que se realizaban manualmente.

Fue la invención del transistor de silicio la que impulsó definitivamente el proceso de automatización y es que una de las principales aplicaciones de este fue la fabricación de computadoras, las cuales crearon la necesidad de interconectarlas entre sí para intercambiar información, dando origen a las redes y consecuentemente, Internet.

1.4.1.2 La automatización en las bibliotecas

La automatización de bibliotecas se entiende como el uso de ciertas herramientas y programas informáticos en las tareas bibliotecarias, las cuales se realizaban manualmente por

el personal bibliotecario. En un comienzo, la automatización comprendía pequeñas aplicaciones introducidas en algún nivel concreto del proceso documental, pero hoy en día afecta a su gestión global.

Son los autores García Melero y García Camarero (1999) los que ofrecen un concepto más detallado y definen a un sistema automatizado de bibliotecas como “un conjunto organizado de recursos humanos que utilizan dispositivos y programas informáticos, adecuados a la naturaleza de los datos, para realizar los procesos y facilitar los servicios que permiten alcanzar el objetivo de la biblioteca: almacenar de forma organizada el conocimiento humano contenido en todo tipo de materiales bibliográficos para satisfacer las necesidades informativas, formativas, recreativas y/o de investigación de los usuarios”.

Es necesario destacar la importancia de los recursos humanos en torno a los sistemas de automatización, ya que son ellos los que aportan su inteligencia y conocimientos. Son comunes los casos en los que un mismo SIGB, utilizado por distintas bibliotecas, ha fracasado en alguna de ellas por la falta de formación del personal o la mala gestión de las actividades y procesos.

Arriola Navarrete y Butrón Yáñez (2008) distinguen cuatro etapas en el desarrollo de la automatización de las bibliotecas:

- La primera comienza en la década de los sesenta en Norteamérica, encabezada por el informe de la Library of Congress (“*Automation and the Library Congress*”, 1963) donde recomendaba automatizar la catalogación. Este informe sentaría las bases para el nacimiento del formato MARC, el cual vería la luz en el año 1966 como MARC I y en 1967 como MARC II.
- La segunda etapa se inicia en los años 70 con la aparición de los microordenadores y los ordenadores personales, los cuales tenían mayor capacidad de almacenamiento. Aparecen nuevos sistemas operativos como diversas implementaciones de UNIX.
- La tercera etapa comienza en la década de los ochenta con el surgimiento de los primeros OPAC y los primeros Sistemas Integrados donde destacan los siguientes: VTLS (1980), NOTIS (1981), CARL (1981) en los EE.UU. En Europa el más conocido fue DOBIS/LIBIS (1979) y en España SABINI (1982) y ABSYS (1990). Destacar la gran demanda de sistemas integrados de bibliotecas, especialmente de las bibliotecas universitarias y públicas, la cual provocó la aparición en el mercado de nuevos proveedores y sistemas de los cuales caben destacar LIBERTAS (1986) y el ya citado ABSYS.
- La cuarta y última etapa abarca desde mediados de los años noventa hasta la actualidad. Se caracteriza principalmente por grandes avances tecnológicos, aparición de estándares e innovaciones diversas como la creación de Internet y el desarrollo de las redes de telecomunicaciones, observadas en los Sistemas de Gestión de Base de Datos (SGBD) y en los Sistemas Integrados de Gestión Bibliotecaria (SIGB) y que han facilitado la automatización bibliotecaria.

Son estos dos últimos autores los que afirman que “la automatización hace muchos años dejó de ser una moda y se convirtió en una necesidad para las bibliotecas. La tecnología

se transformó en una herramienta indispensable para el buen desempeño de los procesos y servicios, así como para el intercambio de información con otras bibliotecas”.

1.4.1.3 Motivos y objetivos de la automatización de bibliotecas

En términos generales se puede afirmar que la principal motivación de la automatización de las bibliotecas es lograr el remplazo de las tareas manuales que se realizan de forma rutinaria e incrementar la productividad y eficiencia mediante el uso de dispositivos tecnológicos.

Para García Melero y García Camarero (1999), los motivos por los que se automatiza una biblioteca son:

- Frenar el colapso del sistema manual, ante el gran volumen de información existente
- Agilizar procesos y optimizar recursos, evitando la repetición de datos y tareas
- Supervisar servicios y así detectar y subsanar posibles fallos o carencias
- Proporcionar información actualizada sobre el funcionamiento de la biblioteca, normalmente a través de un módulo de estadísticas
- Fomentar la cooperación interbibliotecaria compartiendo registros y fondos.

Además, la automatización de las bibliotecas persigue una serie de objetivos (Deharbe, 2012):

- Facilitar y favorecer el acceso a la información tanto a los bibliotecarios como a los usuarios.
- Mejorar, de una forma eficaz y eficiente, los procesos técnicos.
- Ofrecer nuevos servicios y optimizar los ya existentes.
- Compartir recursos con otras unidades de información, favoreciendo así la cooperación.
- Sistematizar los procedimientos y funciones.
- Liberar al personal bibliotecario de la repetición innecesario de procedimientos y de datos.

1.4.2 Sistemas Integrados de Gestión Bibliotecaria (SIGB)

Las bibliotecas iniciaron sus procesos de automatización a partir de los años 80, momento en el que se fue generalizando el uso de los sistemas de computación en los servicios a los usuarios. Esta automatización comenzó por aplicaciones informáticas dirigidas a un servicio específico, por ejemplo, la catalogación, para terminar con los SIGBs, que automatizan de forma integrada todos los procesos y servicios básicos de la biblioteca.

Se entiende como Sistema Integrado de Gestión Bibliotecaria (SIGB) a una herramienta tecnológica que permite automatizar las operaciones bibliotecarias, como el desarrollo de colecciones, la prestación de los servicios bibliotecarios y la administración integral de la biblioteca, evitando la repetición innecesaria de actividades y empleando al máximo los

recursos. Un SIGB integra, en un solo programa informático, un conjunto de aplicaciones específicas o módulos, pensadas para facilitar el trabajo específico en cada una de las tareas de estos centros.

A continuación, se presentan algunas definiciones y posturas que ayudan a observar la evolución del concepto a lo largo de los años:

Moya Anegón (1995) los define como los “sistemas para el proceso automatizado o informático, de información estructurada y no estructurada sobre actividades y documentos, adaptable a la estructura organizativa de la biblioteca”.

Por otra parte, Arriola Navarrete y Butrón Yáñez (2008) los definen como un “conjunto de módulos de aplicación integrados en un solo programa y que comparten una base de datos bibliográficos común”.

Según Flores Vargas (2011), un SIGB es “grupo de programas informáticos (módulos) inter-relacionados, que automatizan múltiples operaciones y funciones bibliotecarias basados en datos centralizados e intercambiables; esto con el objetivo de facilitar la gestión de las actividades llevadas a cabo en la biblioteca”.

Finalmente, Bilal (2014) entiende SIGB como un “Sistema Integrado Bibliotecario Modular” y este a su vez como “un conjunto de componentes o módulos designados utilizando una aplicación software específica para realizar funciones específicas. Los módulos se pueden descomponer según la tarea para la que están diseñados”. Los módulos más comunes son:

- Administración
- Adquisiciones
- Catalogación
- Circulación
- Estadísticas
- OPAC
- Publicaciones periódicas

Independientemente del tipo de sistema que estemos utilizando en la unidad de información, hay una serie de atributos o características que suelen ser comunes entre los distintos SIGBs como:

- Arquitectura cliente-servidor: sirviendo generalmente cualquier navegador web en el “cliente” y dando acceso a interfaces de tipo web; la destinada al usuario, conocida como OPAC (front-end) y la destinada a los bibliotecarios, conocida como Intranet (back-end).
- Compatibilidad con normas y estándares oficiales, como MARC21 o Z39.50.
- Recopilación de información bibliográfica.

- Capacidad para ejecutarse en distintos sistemas operativos, como MS Windows o GNU/Linux.
- Apertura a Internet

1.4.3 Software libre, gratuito, código abierto y propietario

1.4.3.1 Software libre

Para comprender a que nos referimos con el término “software libre” acudimos a la definición de la propia Free Software Foundation, organización creada por Richard M. Stallman en el año 1985 con el propósito de difundir y promocionar el desarrollo y uso del software libre en todas las áreas de la computación.

En esta definición (Free Software Foundation, 2001) se indica que el software libre hace referencia a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. En la misma definición enuncian las llamadas “cuatro libertades del software libre”:

- Libertad 0: libertad para usar el programa, con cualquier propósito y para siempre.
- Libertad 1: libertad para estudiar cómo funciona el programa y poder adaptarlo a nuestras necesidades, para ello es necesario el acceso al código fuente.
- Libertad 2: libertad para distribuir copias del programa, con lo cual puedes colaborar con otras personas.
- Libertad 3: libertad para mejorar el programa y distribuir a terceros las copias de sus versiones modificadas permitiendo a la comunidad de usuarios beneficiarse de esos cambios, para ello también es necesario el acceso al código fuente.

Las cuatro libertades del software libre permiten al usuario hacerse propietario y colaborador al mismo tiempo, esto se debe a que el usuario recibe un producto el cual es útil y que, gracias al principio de colaboración, puede ser utilizado por otros e incluso mejorar el propio software. Es interesante conocer los derechos y responsabilidades que el consumidor debe tomar en cuenta, para ello, Roy Tennant publica en 2007 el *Library Software Manifesto* (Manifiesto del Software para Bibliotecas) donde quedan recogidos todos (véase anexo C)

Es la propia Free Software Foundation la que indica que el termino software libre se refiere a una cuestión de libertad y no de precio para evitar malentendidos, ya que el término “free software” puede crear cierta confusión debido a la ambigüedad de la palabra “free” en inglés, la cual significa tanto “gratis” como “libre”. Si traducimos el término al castellano obtendremos tanto “software gratuito” como “software libre”, los cuales pueden parecer a primera vista lo mismo, pero no lo son.

El “software gratuito”, tal como su nombre indica, se distribuye sin ningún costo, sin embargo, en el “software libre” la palabra libre no implica gratuitidad y aunque este generalmente se distribuye sin costo, es la libertad que lo diferencia la que permite a los usuarios redistribuirlo cobrando tanto como deseen o sea posible. Interdependientemente del precio, la principal diferencia es el acceso al código fuente, el “software libre” lo incluye

y lo pone a disposición del usuario para modificar y redistribuir el software libremente, sin embargo, el “software gratuito” no lo suele incluir y si lo hace, este se encuentra bajo una licencia que especifica la manera en la que debe ser utilizado, modificado y distribuido.

En resumen, solo se considerará como software libre únicamente si los usuarios disfrutan de las cuatro libertades anteriormente mencionadas, si el usuario es libre de redistribuir copias ya estén modificadas o no y ya sea gratis o cobrando una tarifa por su distribución, y distribuyéndolas independientemente de a quién se distribuye o dónde se hace la distribución. Esta redistribución de copias debe incluir tanto las formas ejecutables del programa como el código fuente.

1.4.3.2 Software de código abierto

Tras aclarar a qué llamamos “software libre”, es necesario definir el concepto de “software de código abierto” u “*open source software*”. En este caso acudimos a la definición que da la propia Open Source Initiative, una organización dedicada a la promoción del código abierto y fundada en 1998 por Bruce Perens y Eric S. Raymond.

Esta definición (Open Source Initiative, 1999) indica que el código abierto no es únicamente el acceso al código fuente, sino que debe cumplir con una serie de criterios:

- Libre redistribución: la licencia del software no debe impedir que este sea vendido o distribuido libremente como parte de una distribución mayor que contenga programas de diferentes fuentes
- Código fuente: el programa debe incluir el código fuente y debe permitir la distribución del mismo, así como la forma compilada. En el caso de que el producto no contenga el código fuente, deberá proveer algún medio para obtenerlo, preferiblemente por descarga a través de Internet y sin costo alguno. Además, se rechaza la ofuscación del código fuente⁶.
- Obras derivadas: la licencia debe permitir modificaciones, así como la creación de obras derivadas, ambas podrán distribuirse bajo los mismos términos que la licencia del software original
- Integridad del código fuente del autor: la licencia puede restringir la distribución del código fuente si este ha sido modificado, pero solo si la licencia permite la distribución de parches cuya finalidad sea efectuar dichas modificaciones
- Sin discriminación contra personas o grupos: la licencia no puede restringir el uso del programa a ninguna persona o grupo de personas
- Sin discriminación contra área de iniciativa: la licencia no puede restringir el uso del programa en un campo de actividad concreto
- Distribución de la licencia: los derechos de autor adjuntos al programa se aplicarán a todos aquellos que lo reciben sin necesidad de adquirir una licencia adicional

⁶ Código que se ha hecho ininteligible de forma deliberada.

- La licencia no debe ser específica para un producto: si el programa se usa o distribuye dentro de los términos de la licencia del programa, todas las partes a quienes se redistribuye el programa deben tener los mismos derechos que los del programa original
- La licencia no debe restringir otro software: la licencia no debe imponer ninguna restricción a otros programas que se distribuyan junto con el software con licencia, por ejemplo, la licencia no debe insistir en que todos los demás programas distribuidos en el mismo medio deben ser software de código abierto
- La licencia debe ser neutral a nivel de tecnología: no debe requerirse la aceptación de la licencia por medio de un acceso por clic de ratón o de otra forma específica del medio de soporte del software.

Se pueden observar varias similitudes entre los criterios del software de código abierto y las libertades del software libre. Todos los puntos anteriormente nombrados quedan recogidos en varios tipos de licencias, algunas de ellas son comunes a ambos planteamientos, como por ejemplo las licencias GPL (Licencia Pública General) y LGPL (Licencia Pública General Reducida), ambas asociadas al Proyecto GNU⁷. Para ver un listado comparativo completo de todas las licencias de software libre conocidas hasta el momento acuda a anexo D.

Son estas similitudes las que llevan a considerar los términos “software libre” y “de código abierto” sinónimos, pudiendo utilizar ambas expresiones para referirnos a los dos conceptos indistintamente. Sin embargo, cabe mencionar que es la propia Free Software Foundation la que insiste en considerar ambos conceptos como diferentes:

Nosotros, en el movimiento del software libre, no vemos el ámbito del código abierto como al enemigo; el enemigo es el software privativo, el que no es libre. Pero queremos que la gente sepa que defendemos la libertad, así que no aceptamos que se nos identifique como partidarios del código abierto. (Free Software Foundation, 2001)

Pero más allá de eso las diferencias son casi inexistentes, básicamente el software libre tiene diferencias de índole filosóficas con el Open Source, sin embargo, ambos coinciden en el desarrollo de software de gran calidad y con muchas más libertades de las que el software privativo otorgaría jamás.

1.4.3.3 Software propietario

Por último, el “software propietario”, también denominado “software privativo” o “software de código cerrado” es aquel que no cumple los criterios ni las libertades anteriormente nombradas. Generalmente, su código fuente solo se encuentra a disposición del desarrollador del software y no se permite su modificación, adaptación o redistribución ya que está protegido por copyright llevando la titularidad del dueño o creador. Esto se debe a que el usuario acepta una licencia de uso que le permite utilizar el producto bajo las condiciones anteriormente nombradas, esta licencia se conoce generalmente como

⁷<https://www.gnu.org>

EULA o, en español, CLUF (“*End User License Agreement*” o “Contrato de Licencia para el Usuario Final”).

Esta licencia EULA contiene cláusulas cuyo contenido es completamente opuesto al de los principios y libertades de la Free Software Foundation y la Open Source Initiative. Un claro ejemplo es el siguiente:

2. Restricciones: los usuarios finales y los clientes institucionales no pueden modificar, traducir, descompilar, aplicar ingeniería inversa, revender o redistribuir de ninguna forma ni por ningún medio (electrónico, mecánico, fotocopiado, registrado o de otro tipo) el software EndNote. Y, no pueden traducir, descompilar o aplicar ingeniería inversa a ningún otro componente del software EndNote sin el consentimiento previo por escrito de Clarivate Analytics. (fragmento de la EULA de EndNote)

Dentro del ámbito bibliotecario, adquirir un SIGB propietario es una buena opción puesto que posee algunas ventajas, como el soporte técnico, aunque también supone un costo económico que no todas las bibliotecas pueden cubrir. Casanova Huerta (2010) elabora un cuadro que identifica y concentra las principales diferencias entre el software libre y el propietario:

Software libre	Software propietario
No está limitado a los usuarios, y garantiza las libertades de los usuarios de usar, modificar, copiar y distribuir el software.	Tiene licencias, las cuales están limitadas a los usuarios y en algunos casos son costosas. Estas licencias restringen las libertades de los usuarios a usar, modificar, copiar y distribuir el software.
En el desarrollo de este, pueden intervenir cualquier persona, empresa u organización del mundo. Si bien se genera una gran cantidad de ideas innovadoras, también se posibilita la oportunidad de adecuación de avances tecnológicos en estos productos.	El desarrollo, programación y actualización de este software sólo lo hace la empresa que tiene los derechos.
Los avances, modificaciones y descubrimientos tecnológicos son constantes, y se encuentran en internet de forma gratuita. La principal meta del software libre es compartir los avances tecnológicos con los demás.	El futuro del software que adquirió el usuario solo depende de una empresa comercial.
El usuario no depende de una sola empresa, ya que el software que implementó puede ser mantenido y modificado, ajustándolo a sus necesidades.	En ocasiones las estrategias comerciales suele hacer que los usuarios actualicen su software comercial sin que exista una necesidad verdadera de ello, consiguiendo que el usuario invierta en nuevas licencias, la mayoría de veces innecesarias.

Software libre	Software propietario
Sigue siendo compatible tanto en software como en hardware, no obliga al usuario a cambiar de sistema operativo o equipo, ya que no persigue los mismos fines económicos que el software comercial.	Siendo creado con fines de lucro, las actualizaciones o mejoras al software quedan limitadas bajo las estipulaciones de contrato.

Tabla 4: Principales diferencias entre software libre y propietario. Fuente: Araceli Casanova

2 PLATAFORMAS DE INSTALACIÓN Y PRUEBA DE LOS SIGBS

Una plataforma es un sistema o entorno donde podemos ejecutar, en nuestro caso, aplicaciones de SIGB de software libre.

2.1 AMPdoc

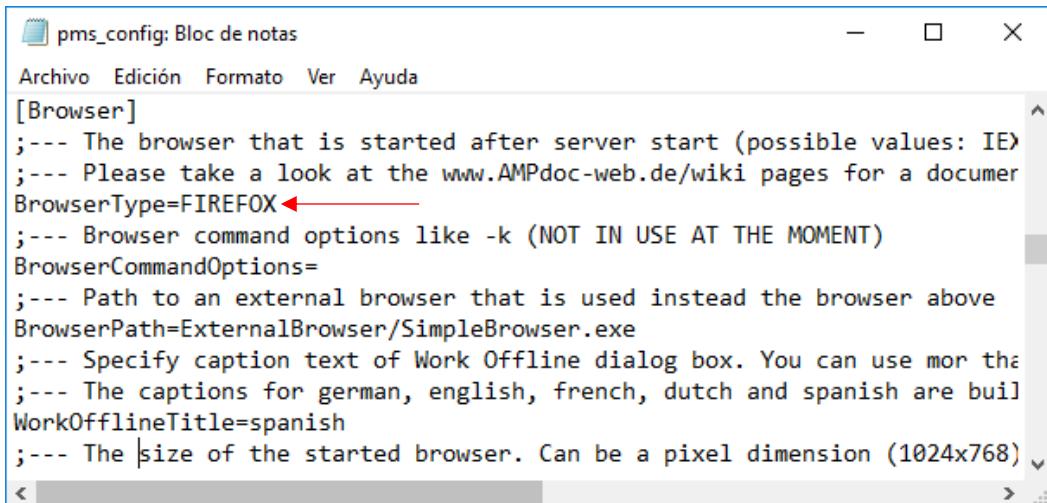
AMPdoc es un paquete de software libre que empaqueta Apache, MySQL, PHP y phpMyAdmin, desarrollado por el profesor Manuel Blázquez Ochando y liberado el día 28 de febrero de 2014. Está basado en Server2Go, un servidor portable y configurable que, al igual que AMPdoc, puede ser utilizado desde cualquier medio de producción, ya sea un pendrive, un disco duro o un portátil, evitando así la necesidad de configurar Apache, MySQL o PHP. AMPdoc se encuentra actualmente en su versión 2.0 (<https://mblazquez.es/ampdoc-2-0/>) y no ha sido actualizado desde 2015.

Está adaptado para la instalación de sistemas de gestión de archivos (Archon, ICA-Atom), Sistemas de Gestión Bibliotecaria (OpenBiblio, PMB y SLiMS), sistemas de gestión de museos (Omeka), junto con otros muchos de interés documental.



Ilustración 2: Pantalla principal de AMPdoc. Fuente: elaboración propia

Su instalación es relativamente sencilla, pues al descargar la distribución AMPdoc (<https://sourceforge.net/projects/ampdoc/>) y descomprimir la aplicación habrá que ejecutar AMPdoc.exe. Podría suceder que el navegador por defecto interrumpiera la ejecución con un mensaje “shutdown”, en este caso será necesario cambiar el navegador por defecto en el fichero “pms_config.ini” la línea “BrowserType=FIREFOX” a “BrowserType=DEFAULT”:



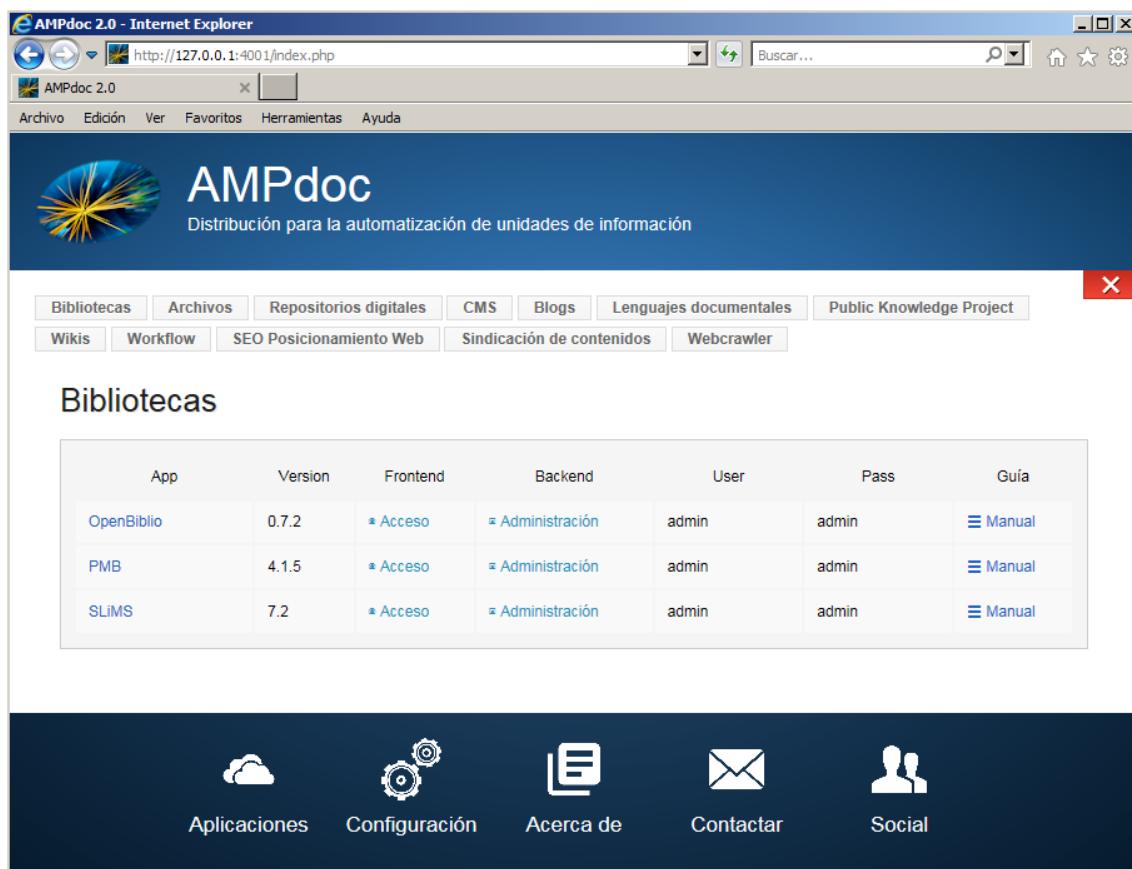
```

pms_config: Bloc de notas
Archivo Edición Formato Ver Ayuda
[Browser]
;--- The browser that is started after server start (possible values: IE>
;--- Please take a look at the www.AMPdoc-web.de/wiki pages for a documen
BrowserType=FIREFOX ←
;--- Browser command options like -k (NOT IN USE AT THE MOMENT)
BrowserCommandOptions=
;--- Path to an external browser that is used instead the browser above
BrowserPath=ExternalBrowser/SimpleBrowser.exe
;--- Specify caption text of Work Offline dialog box. You can use mor tha
;--- The captions for german, english, french, dutch and spanish are buil
WorkOfflineTitle=spanish
;--- The size of the started browser. Can be a pixel dimension (1024x768) ↓

```

Ilustración 3: modificación del fichero de configuración de AMPdoc. Fuente: elaboración propia

Tras cambiar esto, podremos ejecutar AMPdoc con todos los programas y servicios cargados, entre ellos OpenBiblio en su última versión 0.7.2 y PMB en su versión 4.15:



App	Version	Frontend	Backend	User	Pass	Guía
OpenBiblio	0.7.2	* Acceso	* Administración	admin	admin	Manual
PMB	4.1.5	* Acceso	* Administración	admin	admin	Manual
SLIMS	7.2	* Acceso	* Administración	admin	admin	Manual

Ilustración 4: SIGBs disponibles en AMPdoc. Fuente: elaboración propia

Se trata de una aplicación de escritorio que, en principio, no supone ningún riesgo de seguridad para nuestro equipo siempre y cuando se utilice de forma experimental, en modo local y monousuario.

2.2 XAMPP

Se trata de una implementación de WAMP y, al igual que este, es un acrónimo que enumera sus componentes: Apache como servidor web, MariaDB como sistema de gestión de base de datos, PHP como lenguaje de programación y phpMyAdmin como interfaz gráfico del SIGB. La X hace referencia a su disponibilidad para los sistemas operativos MS Windows, GNU/Linux o MacOS. En este trabajo se utilizó la versión 5.6.40 en Windows (<https://www.apachefriends.org/es/download.html>) para instalar OpenBiblio, además, también se puede instalar PMB pero XAMPP requerirá la configuración correcta de módulos como Pear.

Al ejecutar XAMPP se abrirá un panel de control el cual se divide en tres áreas: la de módulos, la de notificación y la de utilidades:

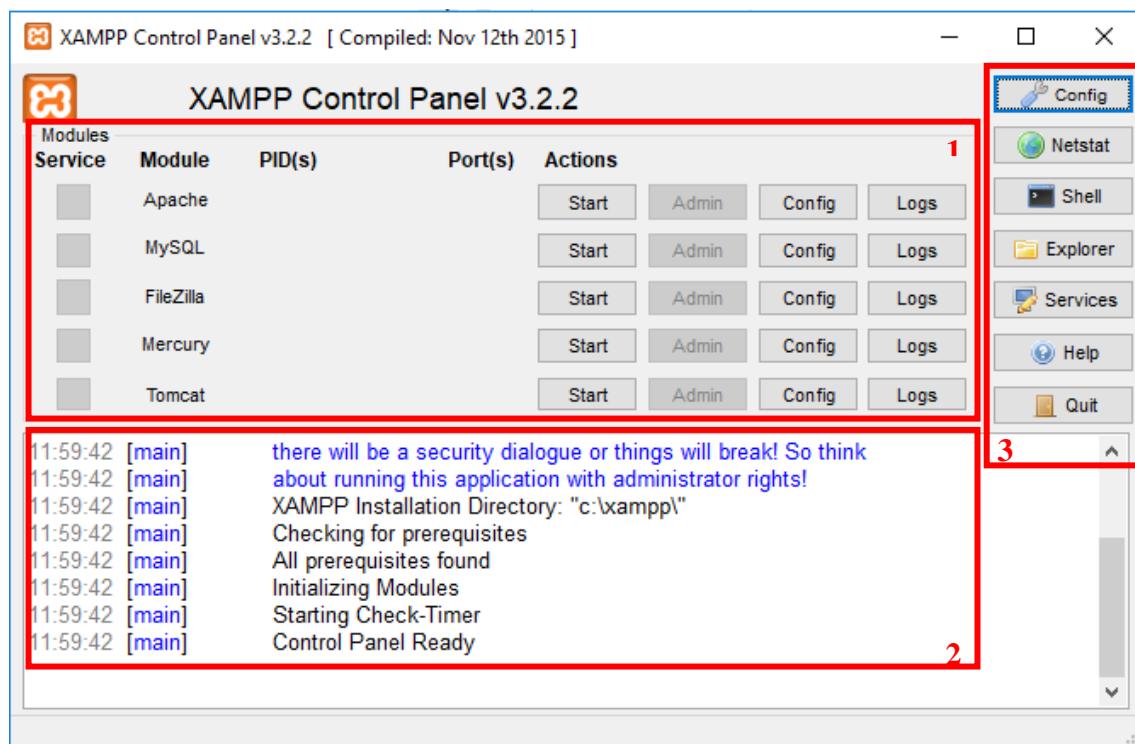


Ilustración 5: Panel de control de XAMPP. Fuente: elaboración propia

- El área de módulos (1) indica los módulos de XAMPP: si se encuentra instalado como servicio, su nombre, el identificador de proceso, puerto utilizado e incluye los botones para iniciar y detenerlos, administrarlos, editar sus archivos de configuración y abrir los archivos de registro de actividad.
- El área de notificación (2) informa del éxito o fracaso de las acciones realizadas.
- El área de utilidades (3) para acceder rápidamente.

Para poder utilizar OpenBiblio será necesario poner en funcionamiento Apache y MySQL, para ello hay que hacer clic en el botón “Start” correspondiente y si el arranque tiene éxito el panel de control mostrará el nombre del módulo con un fondo verde, su identificador de proceso, los puertos abiertos y el botón “Start” ahora se convertirá en “Stop” en caso de querer parar el proceso:

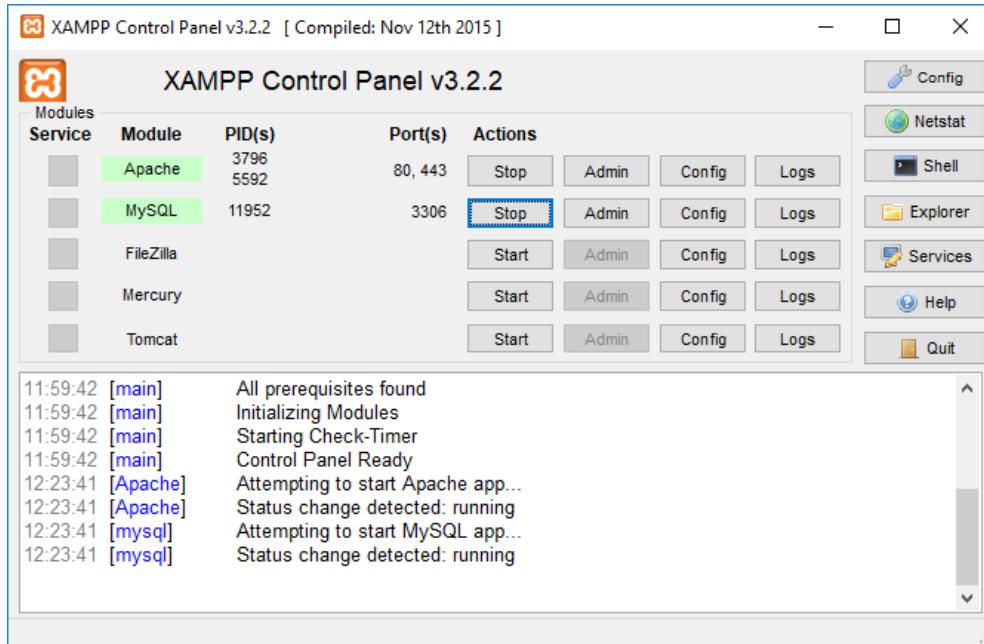


Ilustración 6: Apache y MySQL en funcionamiento. Fuente: elaboración propia

Para comprobar que todo funciona correctamente, hay que abrir cualquier navegador e introducir la dirección <http://localhost/openbiblio> y si todo funciona correctamente aparecerá la pantalla principal de OpenBiblio:

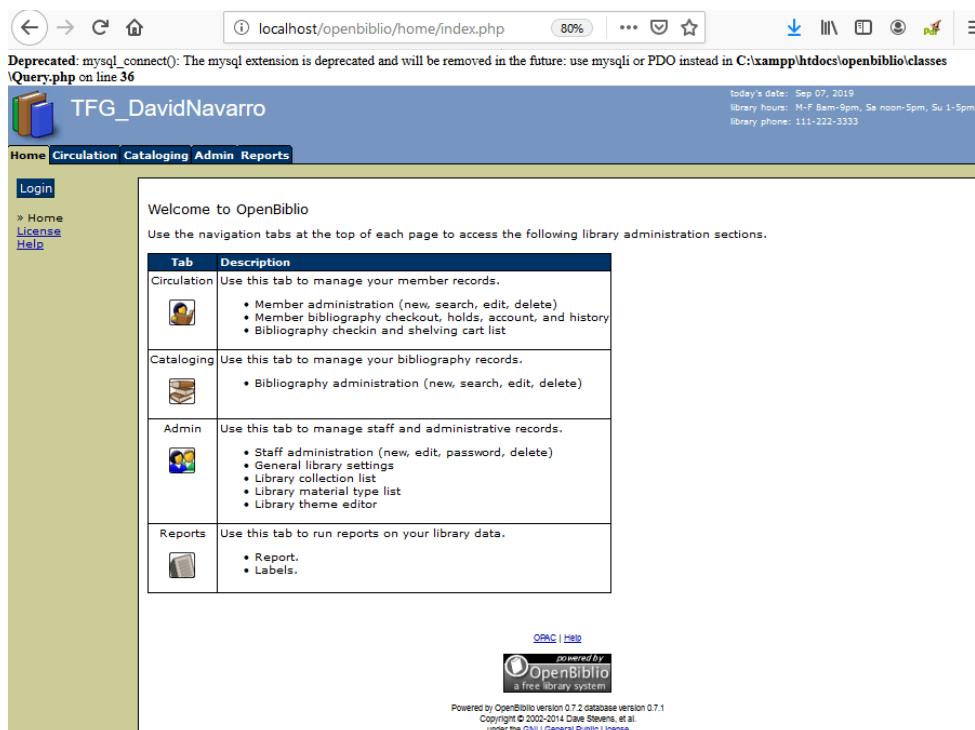


Ilustración 7: Intranet de Openbiblio. Fuente: elaboración propia

Al igual que AMPdoc, XAMPP es una aplicación de escritorio que, en principio, no supone ningún riesgo de seguridad para nuestro equipo siempre y cuando se utilice de forma experimental, en modo local y monousuario.

2.3 LAMP

Se trata de un acrónimo compuesto por las iniciales de sus cuatro componentes: GNU/Linux como sistema operativo, Apache como servidor web, MySQL ó MariaDB como gestor de base de datos y Perl, PHP y Python como lenguajes de programación. Se trata de un entorno profesional y es el más utilizado por las bibliotecas para sus SIGBs.

La plataforma LAMP habitualmente consta de un sistema informático tipo PC (hardware) y alguna de las múltiples implementaciones del Sistema Operativo GNU/Linux, como Ubuntu, Debian, Fedora, SuSE y otras muchas (software). Estas implementaciones permiten escoger, descargar e instalar servidores web (como Apache), Sistemas Gestores de Bases de Datos (como MySQL, mariaDB o PostgreSQL), interfaces gráficos para los SGBDs (como phpMyAdmin), lenguajes de programación (como PHP, Perl, Python), y muchas otras herramientas.

Para escoger la implementación correspondiente deben seguirse las recomendaciones del fabricante del SIGB correspondiente.

Mediante un entorno LAMP se puede instalar cualquiera de los SIGBs mencionados siempre que se tengan los conocimientos necesarios ya que, aplicaciones como Koha y Evergreen que requieren Perl y está última, requiere también el SGBD PostgreSQL el cual precisa muchos más recursos informáticos que MySQL. En nuestro caso utilizamos LAMP para la instalación de PMB:

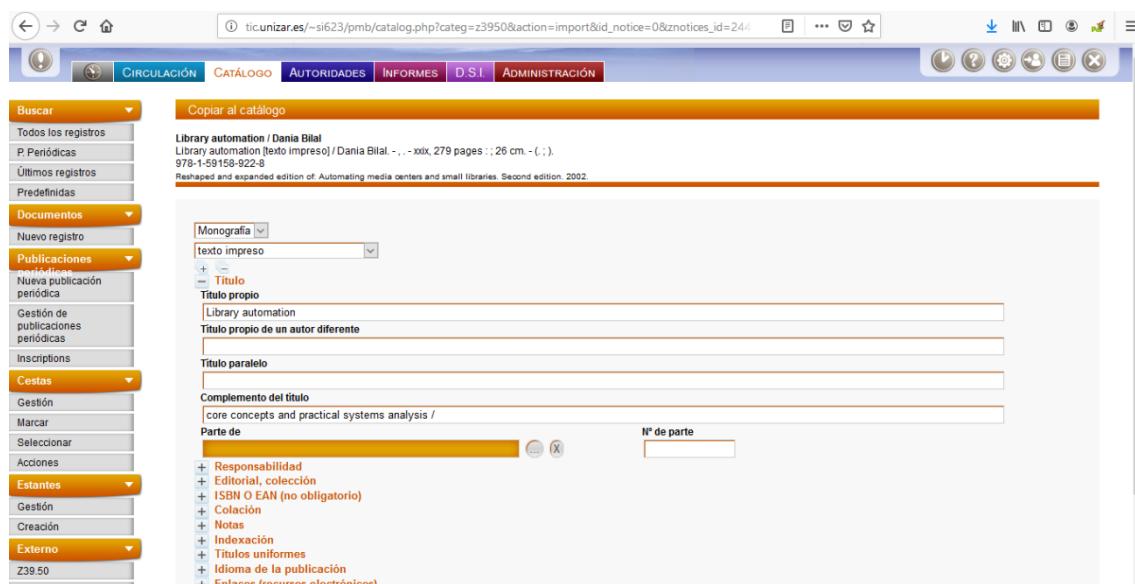


Ilustración 8: Intranet de PMB. Fuente: elaboración propia

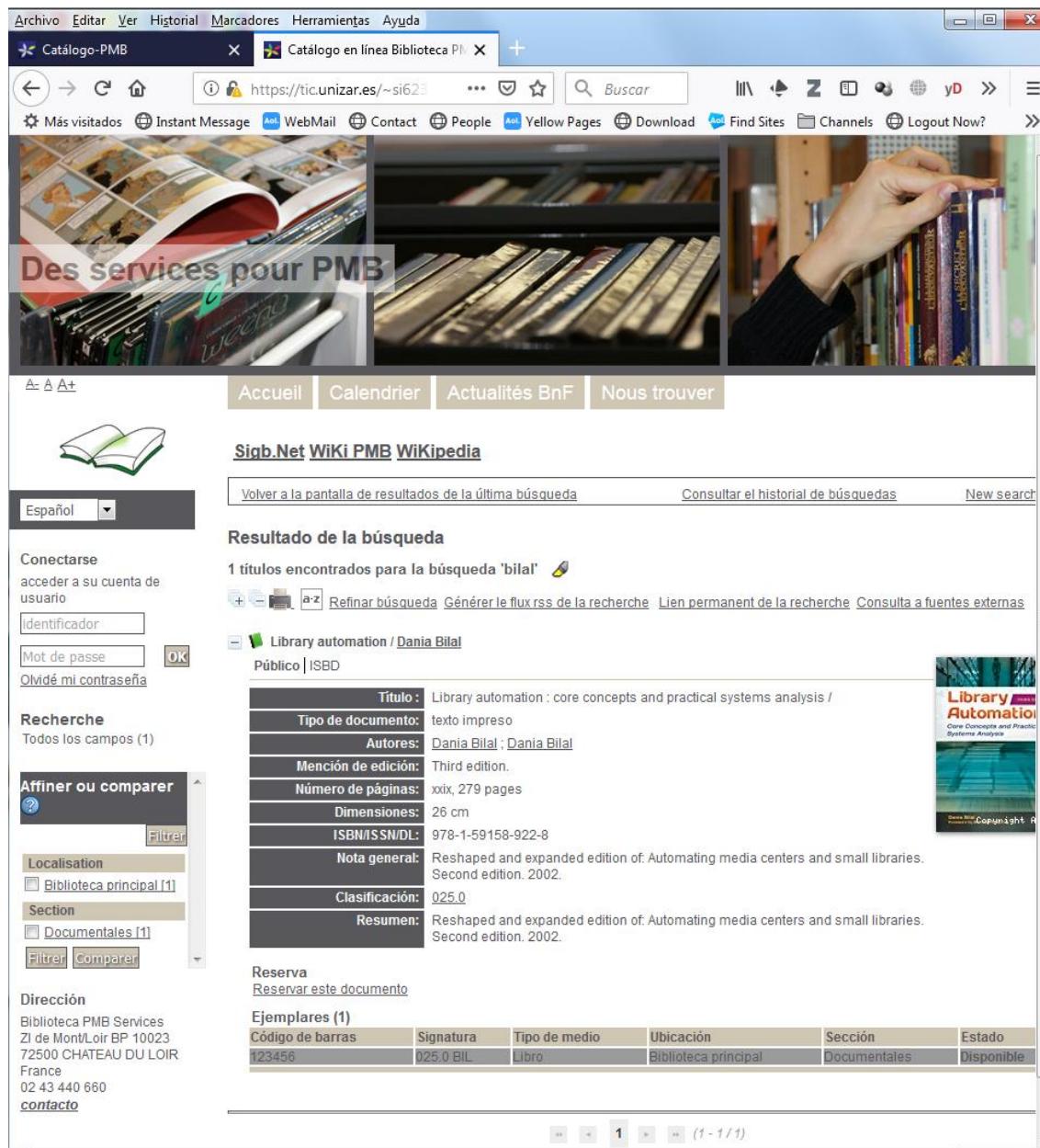


Ilustración 9: OPAC de PMB. Fuente: elaboración propia

LAMP, al igual que AMPdoc y XAMPP, es utilizado en un entorno de enseñanza, pero a diferencia de estos dos, puede y debe utilizarse en un entorno profesional de producción.

La seguridad dependerá completamente del personal responsable del servidor LAMP y de la Intranet de la empresa o institución. Cualquier persona que, con fin docente o profesional, desee disponer de una instalación de una plataforma LAMP completa *deberá contar con los permisos correspondientes por parte de su empresa o institución, y muy recordablemente, conseguir implicar a personal informático y telemático de la misma.*

2.4 Virtualización

Es la creación a través de software, en nuestro caso VMware Player (<https://www.vmware.com>), de una versión virtual de un recurso tecnológico, como lo son los SIGBs.

VMware Player incluye el sistema operativo, los servicios y todas los requisitos instalados y configurados para tan solo tener que poner en funcionamiento la máquina virtual y comenzar a utilizar la aplicación deseada. Se recomienda utilizarlo en entornos de enseñanza y experimentación.

Mediante virtualización se puede instalar Evergreen y Koha, en nuestro caso se instaló Koha-Kobli. En su momento (hasta 2018) este sistema ofrecía una distribución de Koha-Kobli⁸ como aplicación virtual de VMware Player.



Ilustración 10: Intranet Koha-Kobli. Fuente: elaboración propia

⁸Enlace expirado desde comienzos de 2019: <http://kobli.bage.es/descargas-kobli/kobli-1.8-version-virtualizada-con-sistema-operativo/>. Descarga disponible en: <https://archive.org/>

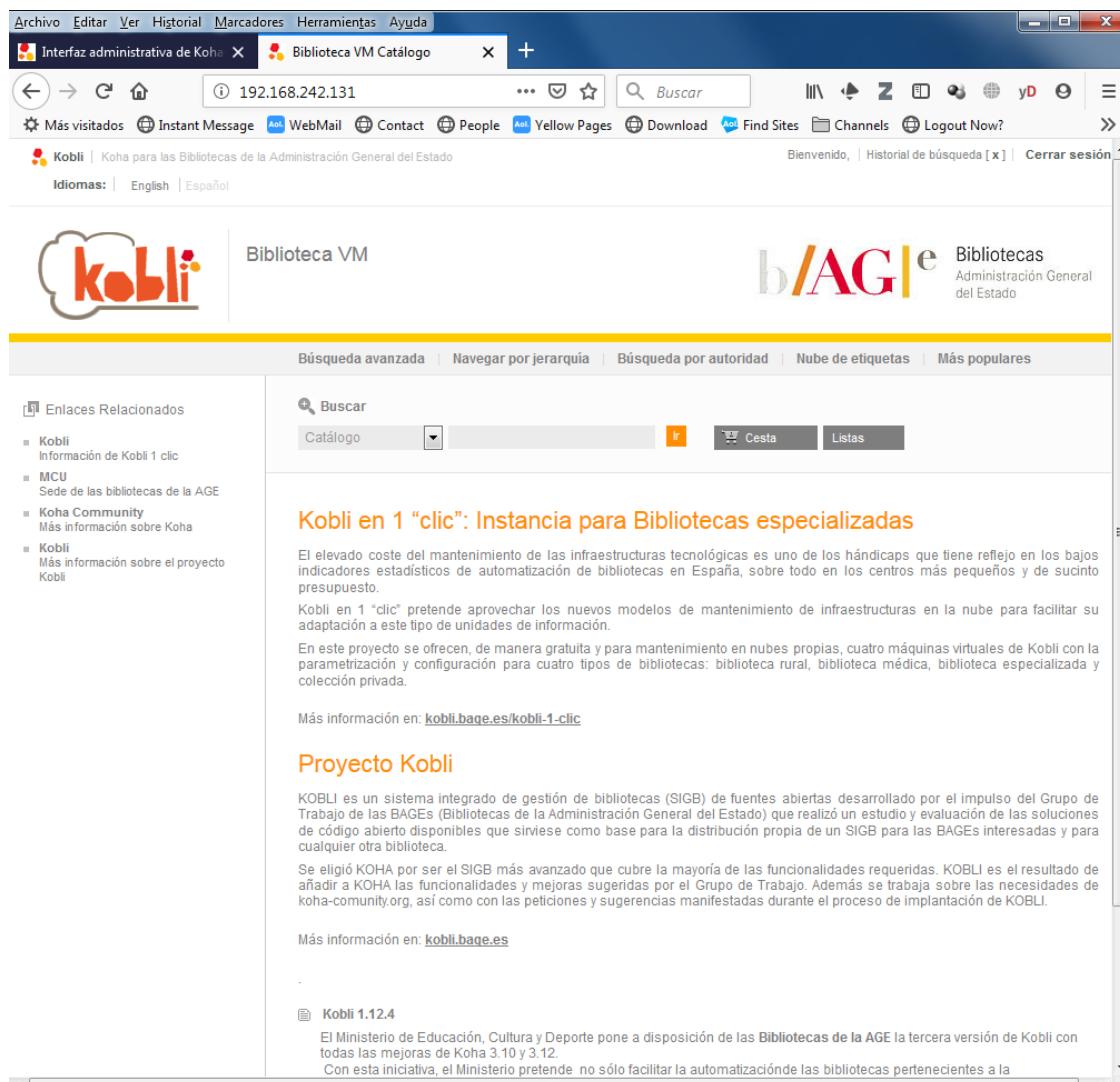


Ilustración 11: OPAC de KOHA-Kobli. Fuente: elaboración propia

En cuanto seguridad, no debería haber problemas si solo se permite el acceso en modo local, de una forma similar a XAMPP.

2.5 Otros medios

En caso de que en un entorno de enseñanza o profesional se quiera experimentar con software libre sin pasar por el trabajo que supone su instalación, existen “demos online” de los mismos. En este trabajo, los programas no instalados de software libre fueron estudiados mediante las correspondientes demos disponibles en la web:

- **Demo ABCD**
 - Intranet: <http://www.abcdwiki.net/ABCD/?lang=es>
 - Esta demo no dispone de OPAC
 - Se encuentra en su versión 1.5.2

- Disponible en inglés, castellano, francés y portugués
- Usuario: adm – Contraseña: adm
- **Demo Evergreen**
 - Intranet: <https://demo.evergreencatalog.com/eg/staff/>
 - OPAC: <https://demo.evergreencatalog.com/eg/opac/home>
 - Se encuentra en su versión 3.2.5
 - Disponible en inglés
 - Usuario: admin – Contraseña: demo123
- **Demo Koha**
 - Intranet: <https://koha-community.org/demo/>
 - OPAC: <http://koha.adminkuhn.ch/>
 - Se encuentra en su versión 19.05
 - Disponible en inglés, francés, italiano y alemán
 - Usuario: demo – Contraseña: demo
- **Demo Meran**
 - Intranet: <https://demo.meran.unlp.edu.ar/meran/auth.pl>
 - OPAC: <http://demo.meran.unlp.edu.ar/>
 - Version sin especificar
 - Disponible en castellano
 - Usuario: demo – Contraseña: demo

3 ANÁLISIS Y DISCUSIÓN DE RESULTADOS

Tras realizar una exhaustiva búsqueda se recuperaron un total de treinta y tres (33) aplicaciones y se procedió a comprobar si se trataba de software libre o gratuito.

El mercado de los SIGBs se encuentra en constante evolución, y nuevas aplicaciones surgen cada día, por lo tanto, es necesario recalcar que esta lista de no es definitiva y está abierta a ser actualizada y modificada en cualquier momento.

A continuación, una tabla que recoge aquellas aplicaciones recuperadas, el tipo de software que la conforma, si está disponible en español, su última actualización y su URL.

SIGB	Software	Disponible en español	Última actualización	URL
ABCD	Libre	Si	Junio, 2019	http://abcd.net-cat.be/
Aguapey ⁹	Gratuito	Si	No disponible	http://www.bn.m.me.gov.ar/res_federales/aguapey/
Avanti MicroLCS	Libre	No	No disponible	No disponible
BIBLIO	Gratuito	Si	Febrero, 2018	http://biblio.vzpla.net/?i=1
Biblioteca 2000	Gratuito	Si	Enero, 2011	https://smdani.com/biblioteca-2000-descargar/
BiblioteQ	Libre	No	Marzo, 2019	https://biblioteq.sourceforge.io/
Biblivre	Libre	No	Diciembre, 2016	http://www.biblivre.org.br/index.php

⁹ Se distribuye gratuitamente solo a aquellas instituciones que participan del Programa Bera (Programa Nacional de Bibliotecas Escolares de la República Argentina).

SIGB	Software	Disponible en español	Última actualización	URL
CaMPI	Libre	Si	Noviembre, 2011	https://web.archive.org/web/20150705162747/http://www.campi.uns.edu.ar/
Catalis	Libre	Si	Junio, 2005	http://in-mabb.criba.edu.ar/catalis/
Colibrí ¹⁰	Libre	Si	No disponible	https://mblazquez.es/colibri/
DSpace	Libre	No	Junio, 2018	https://duraspace.org/dspace/
Emilda	Libre	No	Junio, 2005	https://source-forgue.net/projects/emilda/
Evergreen	Libre	Si	Agosto, 2019	https://evergreen-ils.org/
FOLIO ¹¹	Libre	No	Mayo, 2019	https://www.folio.org/
GBwinNet	Gratuito	Si	Noviembre, 2018	https://gbwin-net.com/
GLIBMS	Libre	No	Abril, 2015	https://source-forgue.net/projects/glibs/
GNUteca ¹²	Libre	Si	Julio, 2005	http://www.solis.com.br/gnuoteca
Greenstone ¹³	Libre	Si	Noviembre, 2016	http://www.greenstone.org/
Invenio	Libre	No	Marzo, 2019	https://invenio-software.org/
Koha ¹⁴	Libre	Si	Agosto, 2019	https://koha-community.org/

¹⁰ El software Colibrí creado por Manuel Blázquez no ha sido publicado, ya que el autor solicita una cantidad de “likes” en un post de Facebook, la cual no ha sido alcanzada.

¹¹ FOLIO se encuentra actualmente bajo desarrollo, pero once bibliotecas plantean implementarlo en 2019. Es considerado como la siguiente generación de los sistemas bibliotecarios ya que no solo ofrece las funciones de un SIGB clásico, también brinda la oportunidad de integrar los requisitos futuros de la biblioteca en el sistema.

¹² Existe una versión traducida al castellano en el gettext del paquete. Para adquirir Gnuteca hay que solicitarlo a través de su página oficial junto con un servicio de soporte que no es gratuito.

¹³ Aunque Greenstone cumple con los criterios establecidos, este se descarta debido a que no es un SIGB en sí, sino una herramienta para la creación y gestión de colecciones de bibliotecas digitales.

¹⁴ Existe Koha-Kobli, un fork de Koha que añade funcionalidades al software original y está completamente en castellano. No ha sido añadido a la lista debido a que no se considera una aplicación en sí y porque su desarrollo y mantenimiento concluyó definitivamente a comienzos del año 2019.

SIGB	Software	Disponible en español	Última actualización	URL	
Librarika	Gratis	No	Mayo, 2018	https://librarika.com/	
NewGenLib	Libre	No	Mayo, 2015	http://www.verussolutions.biz/web/	
Next-L Enju	Libre	No	Enero, 2018	https://foss4lib.org/package/next-l-enju	
Meran	Libre	Si	Julio, 2017	https://cespi.unlp.edu.ar/meran	
OPALS	Libre	No	Mayo, 2012	https://opal-sinfo.net/	
Open Marco-Polo	Marco-Polo	Libre	Si	Mayo, 2008	http://marco-polo.uner.edu.ar/
OpenBiblio	Libre	Si	Agosto, 2014	http://obiblio.sourceforge.net/	
OpenOpac	Libre	Si	Abril, 2003	No disponible	
OtomiGenX	Libre	No	Junio, 2014	https://foss4lib.org/package/otomigenx	
PHPMyLibrary	Libre	No	Abril, 2013	https://sourceforge.net/p/phpmylibrary/wiki/Home/	
PMB	Libre	Si	Marzo, 2019	https://www.sib.net/	
SLiMS	Libre	No	Mayo, 2014	https://slims.web.id/web/	
WEBLIS	Libre	No	Agosto, 2018	http://www.unesco.org/new/en/communication-and-information/information-information-society/open-source-and-low-cost-technologies/information-processing-tools/cdsis-database-	

SIGB	Software	Disponible en español	Última actualización	URL
				soft-ware/weblis/

Tabla 5: Lista de SIGBs recuperados

Tan solo 6 de las 33 aplicaciones recuperadas cumplieron con los requisitos establecidos para poder ser analizadas. Además, pese a no cumplir con el requisito de ser una aplicación de software libre, BIBLIO fue analizada debido a que se trata de un SIGB de gran facilidad de instalación y uso que puede llegar a ser de gran ayuda a muchos bibliotecarios.

A continuación, un análisis de las aplicaciones desde su desarrollo, sus generalidades, requisitos de instalación y funcionalidades.

3.1 ABCD



ABCD es el acrónimo utilizado para designar un paquete de software para la automatización de bibliotecas y centros de documentación. Es español se traduce como “Automatización de Bibliotecas y Centros de Documentación”, el acrónimo es el mismo para el francés (Automation des Bibliothèques et Centres de Documentacion) o el portugués (Automatização das Bibliotecas e dos Centros de Documentação).

Su versión 1.0 fue liberada en 2009 en Brasil tras dos años de desarrollo gracias a la cooperación entre BIREME (Centro Latinoamericano y del Caribe de Información en Ciencias de la Salud) y el Vlaamse Interuniversitaire Raad (Consejo Interuniversitario Flamenco, Bélgica). Es un sistema integrado que forma parte de la familia de productos ISIS, un paquete de software desarrollado y mantenido por la UNESCO y diseñado en particular para bibliotecas pequeñas y medianas de países en vías de desarrollo.

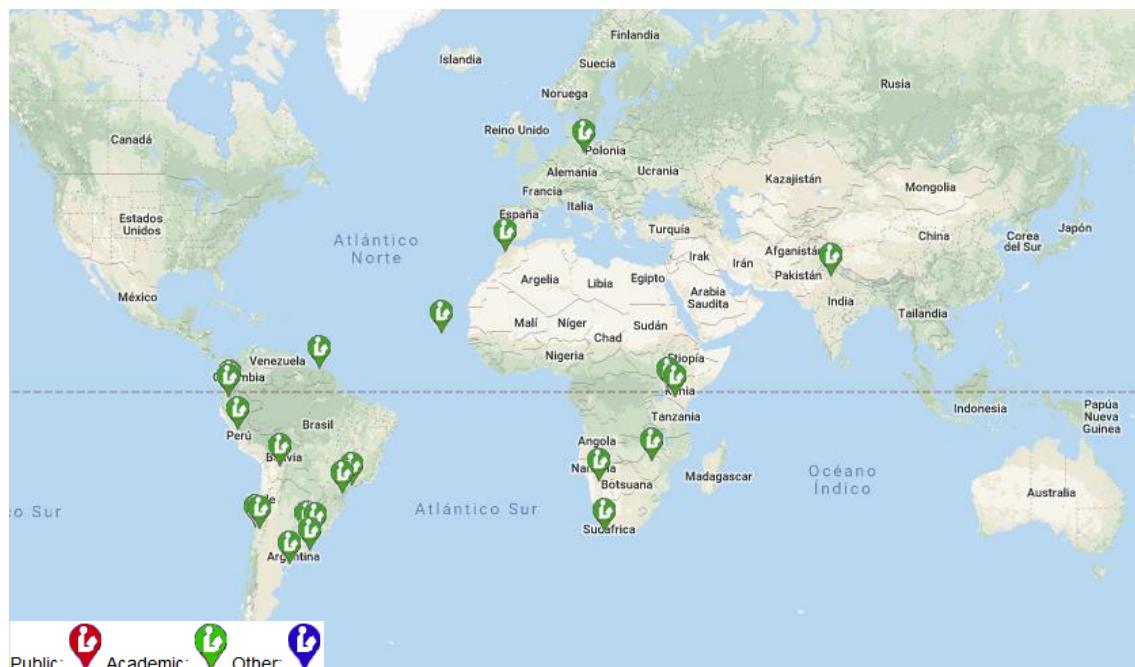


Ilustración 12: Mapa de instalaciones de ABCD. Fuente: Library Technology

Su propio nombre indica la ambición del producto: no solo proveer servicio para bibliotecas sino también para centros de documentación, archivos e incluso museos. Es un sistema que destaca por su flexibilidad y versatilidad, los criterios bajo los cuales fue desarrollado. Se puede apreciar esta flexibilidad en el hecho de que, cualquier estructura bibliográfica puede ser manejada por el software. Se presenta como software que incorpora los estándares actuales más importantes como MARC21, UNIMARC o CEPAL. Además, ABCD viene con su propio servidor OAI-HP que permite la recolección de registros por parte de otros servicios web que no estén basados en ISIS.

ABCD es considerado como un “paquete” de softwares debido a que está formado por un conjunto de módulos independientes entre sí, que pueden cooperar totalmente, pero que también pueden existir de manera aislada. Estos quedan recogidos en la “suite” o módulo central de ABCD, el cuál comprende todos los módulos necesarios para automatizar una biblioteca de pequeño tamaño. Además, permite diseñar la estructura y los componentes de su página principal de tal forma que es completamente adaptable a las necesidades de la unidad de información. Tiene una arquitectura cliente-servidor, característica de la mayoría de los SIGBs.

a) Generalidades

Características	Descripción
Nombre de la aplicación	ABCD (Sistema de Automatización de Bibliotecas y Centros de Documentación)
Versión	2.1 ¹⁵ (Junio, 2019)
País de origen	Brasil
Idiomas que maneja	Multilingüe (inglés, español, francés y portugués)
Creador	BIREME (Centro Latinoamericano y del Caribe de Información en Ciencias de la Salud) y Vlaamse Interuniversitaire Raad (Consejo Interuniversitario Flamenco, Bélgica).
Licencia	GNU LGPL (Lesser General Public License)
Página web	http://abcd.netcat.be/
Página de descarga	http://abcd.netcat.be/files/downloads.html
Soporte	Manuales y otros documentos: http://abcd.netcat.be/files/documents.html Wiki: http://abcdwiki.net/wiki/es/index.php?title=P%C3%A1gina_principal

Tabla 6: Generalidades de ABCD

¹⁵ La versión 3.0 está siendo desarrollada por la UCI (Universidad de las Ciencias Informáticas) en Cuba.

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux, MacOS y MS Windows.
Servidor	Apache
Lenguaje de programación	PHP, Java
Sistema Gestor de Base de Datos	MySQL

Tabla 7: Requisitos de instalación de ABCD

c) Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	Si
Módulo de catalogación	Si
Módulo de circulación	Si ¹⁶
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	Si
Módulo de gestión de usuarios	Si
Módulo de reservas	Si

Tabla 8: Funcionalidades de ABCD

3.2 BIBLIO



BIBLIO es una aplicación de software gratuito destinada a la gestión de bibliotecas de pequeño tamaño de centros y/o instituciones educativas. Permite una sencilla gestión y administración de estas: consultas bibliográficas, altas de libros, gestión de préstamos, etc.

Pese a que se trata de una aplicación gratuita y no libre, ya que no da acceso a su código fuente, esta se ha añadido debido a este trabajo está destinado a servir de ayuda y BIBLIO es uno los sistemas integrados que pueden ser de gran utilidad para

¹⁶ ABCD contiene un segundo módulo de circulación conocido como EmpWeb, el cuál es incompatible con el módulo simple. EmpWeb ofrece una serie de mejoras como la posibilidad de hacer reservas o la función “mi biblioteca”, a través de la cual los usuarios pueden comprobar en línea información personal, préstamos, etc.

aquellos bibliotecarios que estén a cargo de una biblioteca de pequeño tamaño y no necesiten más funcionalidades de las que ofrece este SIGB. Se trata de una aplicación de escritorio, cuya instalación es muy simple y no requiere de un entorno web 2.0.

Fue desarrollado por José A. Graña y su primera versión fue lanzada en diciembre de 1996, desde entonces ha sido revisado, mejorado y actualizado en numerosas ocasiones, encontrándose ya en su versión 8.26. Actualmente solo se encuentra disponible en castellano, gallego e inglés. Destaca por su sencillez e inmediatez puesto que no se necesita ningún tipo de registro para utilizarlo, tan solo su instalación en cualquier ordenador con la una versión MS Windows-95 o superior.

Existe una versión comercial de BIBLIO la cual necesita un número de registro y está pensada para entidad privadas y comerciales. La única diferencia es que en la versión comercial aparecerá el nombre de la persona o entidad que lo ha registrado mientras que en la versión gratuita no aparecerá nada.

BIBLIO es una aplicación shareware, es decir, puede ser copiada, distribuida y usada durante el tiempo necesario antes de registrarla y/o adquirir la versión comercial. Es totalmente gratuita si está destinada a centros educativos.

Permite importar datos desde otro programa de gestión de bibliotecas, pero no en todos los casos pues dependerá de la compatibilidad de la base de datos con la estructura de la base de datos de BIBLIO, se recomienda ponerse en contacto con el creador del programa a través de su correo electrónico.

a) Generalidades

Características	Descripción
Nombre de la aplicación	BIBLIO
Versión	8.26 (Febrero, 2018)
País de origen	España
Idiomas que maneja	Multilingüe (Español, gallego e inglés)
Creador	José A. Graña
Licencia	Shareware
Página web	http://biblio.vzpla.net/?i=1
Página de descarga	http://biblio.vzpla.net/desSP.html
Soporte	Correo electrónico para dudas: joseagrana@wanadoo.es Manual BIBLIO: https://docplayer.es/8066758-Manual-de-acceso-rapido-biblio.html

Tabla 9: Generalidades de BIBLIO

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux y MS Windows
Servidor	—
Lenguaje de programación	VisualBasic
Sistema Gestor de Base de Datos	Jet (MS Access)

Tabla 10: Requisitos de instalación de BIBLIO

c) Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	—
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	—
Módulo de publicaciones periódicas	—

Tabla 11: Funcionalidades de BIBLIO

3.3 Evergreen



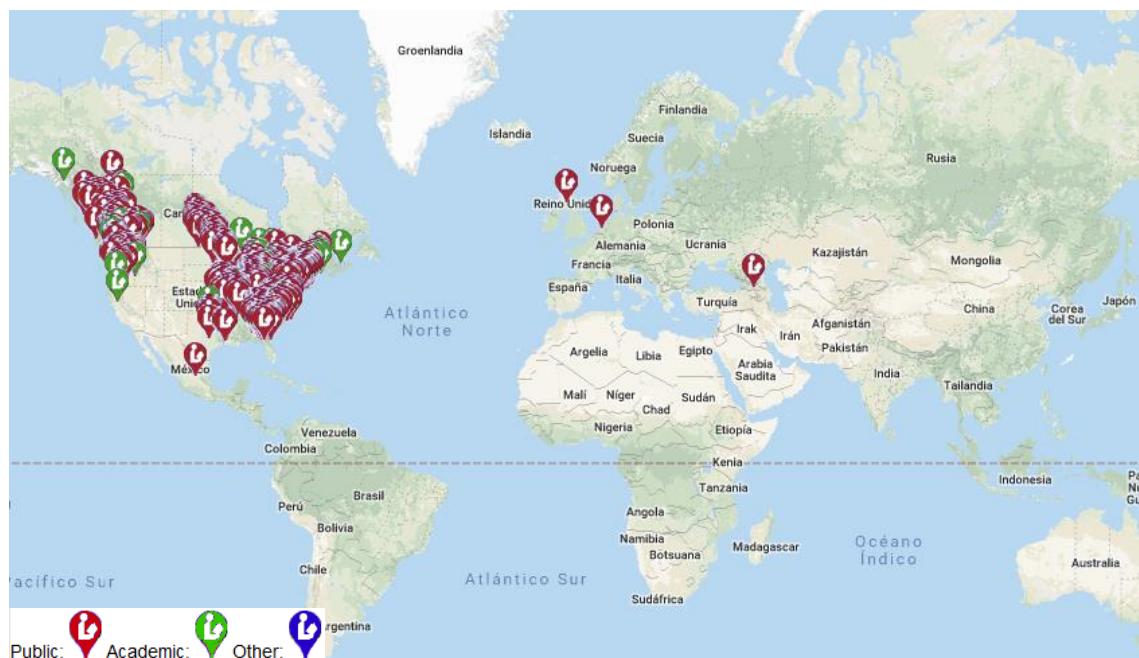
Entre la oferta existente de SIGBs de software libre, Evergreen es uno de los grandes protagonistas teniendo en cuenta que es uno de los más populares a la vista del número de bibliotecas en el que es utilizado como detalla Breeding (2015). Está diseñado para su uso en grandes consorcios de bibliotecas.

El plan para desarrollar Evergreen fue anunciado por primera vez en junio de 2004 por Lamar Veatch, jefe del Servicio de Bibliotecas Públicas de Georgia (GPLS, por sus siglas en inglés), pero no comenzó a funcionar hasta 2006. GPLS tomó la importante decisión de desarrollar un software compartido de automatización de bibliotecas para aquellas instituciones ubicadas en el Estado de Georgia y que formaran parte de PINES (Public Information Network for Electronic Services) o la red de recursos compartidos de los sistemas de bibliotecas de Georgia, el cuál fue fundado en 1998 y llegó a recoger 266 de las

408 bibliotecas del estado. Estas utilizaban Unicorn como SIGB, siendo Sirsi (posteriormente SirsiDynix) la compañía que les daba soporte hasta que finalizara el contrato en junio de 2005. Fue entonces cuando se tomó la decisión de migrar a otro sistema: por una parte, debido a que Unicorn no satisfacía las necesidades de la GPLS, y por otra parte se llegó a la conclusión de que desarrollar un sistema propio adaptado a esas necesidades resultaría mucho menos costoso que seguir pagando el soporte de Sirsi.

El trabajo de desarrollo de Evergreen comenzó a mediados de 2004 como un proyecto de dos años para crear un SIGB para el consorcio de PINES: fue completado finalmente en septiembre de 2006, cuando finalizó la migración desde Unicorn a Evergreen. Como resultado de lo que se aprendió sobre el uso de Unicorn y las necesidades que este no pudo suplir, Evergreen fue desarrollado para que fuera escalable, robusto, tolerante a fallos y, por supuesto, código abierto:

- Escalable: el software puede implementarse en bibliotecas de diferentes tamaños y agregar capacidad es relativamente sencillo. En el momento en que se tomó la decisión de migrar de Unicorn, la actualización del servidor para este sistema habría costado 1,5 millones de dólares, una cifra mucho superior de lo que costó el desarrollo de Evergreen a la GPLS. Evergreen puede ser actualizado agregando servidores de cualquier fabricante y puede ejecutarse en un ordenador portátil.
- Robusto: Evergreen fue diseñado para poder seguir funcionando en situaciones extremas. En una ocasión una excavadora accidentalmente cortó por completo la red de PINES, aun así, el personal de Evergreen permitió que las bibliotecas siguieran realizando sus labores incluso estando desconectadas de las bases de datos centrales.
- Tolerante a fallos: en caso de fallos en servidores, Evergreen seguirá funcionando. Esto se debe al uso de servidores redundantes, permitiendo que si uno falla los otros siguen funcionando.



Los desarrolladores de Evergreen abandonaron en 2007 el GPLS para fundar su propia empresa en torno al software, Equinox Software, que proporciona soporte, desarrollo, migración, capacitación y la consultoría de Evergreen.

Evergreen está programado en lenguaje Perl, aunque algunas partes fueron reprogramadas para optimizarlas en lenguaje de programación C. Utiliza una arquitectura cliente-servidor un tanto peculiar, pues, en la parte del servidor, requiere un equipo con sistema operativo GNU/Linux, servidor Apache, pero utiliza PostgreSQL en lugar de MySQL.

a) Generalidades

Características	Descripción
Nombre de la aplicación	Evergreen
Versión	3.3.3 (Agosto, 2019)
País de origen	Estados Unidos
Idiomas que maneja	Multilingüe (árabe, armenio, checo, inglés, francés, portugués y español).
Creador	Servicio de Bibliotecas Públicas de Georgia
Licencia	GNU GPL (General Public License)
Página web	https://evergreen-ils.org/
Página de descarga	https://evergreen-ils.org/egdownloads/
Soporte	Blogs: https://evergreen-ils.org/communicate/blog/ https://planet.evergreen-ils.org/ Listas de discusión: https://evergreen-ils.org/communicate/mailing-lists/ Manuales: http://docs.evergreen-ils.org/ Wiki: https://wiki.evergreen-ils.org/doku.php

Tabla 12: Generalidades de Evergreen

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux, MacOS y MS Windows.
Servidor	Apache
Lenguaje de programación	Perl, C
Sistema Gestor de Base de Datos	PostgreSQL

Tabla 13: Requisitos de instalación de Evergreen

c) Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	Si
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	Si

Tabla 14: Funcionalidades de Evergreen

3.4 Koha



Koha, junto a Evergreen, es considerado como uno de los SIGBs más representativos del software libre, es el más veterano y, posiblemente, el que mayor éxito y difusión ha tenido de entre todos los nombrados en este trabajo.

Fue creado en 1999 por Katipo Communications, una compañía especializada en el diseño y desarrollo de proyectos de código abierto, cuando el Horowhenua Library Trust decidió dejar atrás un sistema de 12 años de antigüedad, Catalyst, para migrar a uno nuevo. La biblioteca fue consciente del alto coste que suponía adquirir un producto comercial. Ante la imposibilidad de financiarlo decidieron contactar con Katipo Communications para desarrollar un nuevo sistema basado en la web. El proyecto se desarrolló durante 1999 y el 1 de enero del 2000 ya estaba lista la primera versión de Koha. Katipo Communications y la HLT decidieron liberar el sistema bajo la licencia GPL, para que otras bibliotecas pudieran utilizarlo, así como ayudar en su desarrollo.

Fue en enero del año 2000 cuando se realizó la primera instalación de Koha, recibiendo a su vez dos reconocimientos: el 3M a la innovación en Bibliotecas y el interactivo ANZ (Categoría comunitaria/no lucrativa). Koha fue adoptada por otras bibliotecas en Nueva Zelanda, siendo Katipo Communications la compañía que ofrecía el soporte. En 2001, la Nelsonville Public Library (Ohio, Estados Unidos) mostró gran interés por Koha y decidió sustituir su entonces sistema Spydus. Esta biblioteca atendía, a través de una sede central y seis bibliotecas sucursales, a un total de 62.000 habitantes y su fondo estaba formado por más de 250.000 volúmenes. Koha, por aquel entonces, había sido implementado en bibliotecas de menor tamaño y no tenía la capacidad suficiente para gestionar una biblioteca de ese calibre.

La Nelsonville Public Library apoyó el desarrollo de Koha para adaptarlo a su biblioteca, esto supuso añadirle nuevas funcionalidades como la catalogación en formato MARC 21, el protocolo Z39.50 y los protocolos SIP2 (Estándar Interchange Protocol) y NCIP (National Information Standards Organization Circulation Interchange Protocol) necesarios para las máquinas de autopréstamo. A su vez, el Crawford County Federated Library System agregó a Koha la base de datos contextual Zebra, la cual aumento considerablemente la velocidad de búsquedas y la capacidad de Koha, permitiéndole ser utilizado en bibliotecas de mayor tamaño.

En 2002, Paul Poulain, patrocinado por la Nelsonville Public Library de Ohio, comenzó a agregar las nuevas características al sistema. En ese mismo año Poulain fundaría Biblibre, compañía francesa con sede en Marsella, que daría soporte técnico y desarrollo a Koha.

En 2005 Joshua Ferraro, un administrador de sistemas que lideró el proyecto de migración de algunas bibliotecas a Koha, fundaría Liblime cuyo fin sería idéntico a Biblibre, pero en este caso su sede se localizaría en Maryland, Estados Unidos. Liblime se convirtió en el principal impulso de Koha al comprar Katipo Communications y adquiriendo así activos relacionados con el programa: 108 contratos de clientes con 160 organizaciones que representan a más de 500 bibliotecas individuales, derechos de autor sobre el código fuente original de Koha, dominios de Internet, marcas registradas, etc.

Tras esta importante adquisición, Liblime tomaría decisiones cuestionables y llenas de polémica. En 2009 anunció la creación de una nueva versión de Koha y, contrariamente al software original, este sería de tipo propietario: “Enterprise Koha”. Sus usuarios tendrían prioridad a la hora de recibir actualizaciones y nuevas prestaciones. En un principio esta decisión no debería ser cuestionada pues muchas compañías de software libre desarrollan la versión de pago de sus aplicaciones de software libre, pero Liblime decidió que sus programadores dejaran de trabajar en el Koha original para centrarse totalmente en esta nueva versión.

Todo este proyecto quedó paralizado cuando, en marzo 2010, la empresa Progresive Technology Federal Systems (PTFS Inc.) anunció la compra de Liblime. Cabe destacar que esta empresa es ArchivalWare (desde 2015, Knowvation) y da soporte a otros SIGBs aparte de Koha, como el ya mencionado Evergreen. En un principio, PTFS Inc. no mostró interés por un Koha abierto, sino una evolución del Enterprise Koha hacia un sistema para bibliotecas universitarias. Esto no tuvo una buena respuesta por parte de la comunidad y empresas como Equinox o Biblibre comenzaron a ofrecer servicios de soporte profesional a Koha. PTFS en un intento por recuperar la confianza de la comunidad mostró su compromiso por un Koha abierto.

Liblime continuó funcionando como división de PTFS y, en noviembre de 2011, la disputa en la comunidad de Koha se profundizó cuando Liblime solicitó al Ministerio de Desarrollo Económico de Nueva Zelanda registrar el nombre de “Koha” como marca de su propiedad, irónicamente “koha” significa “donación” o “regalo” en maorí. El Horowhenua Library Trust, que originalmente había desarrollado Koha, se opuso completamente a la propuesta de Liblime con la siguiente declaración:

“El Horowhenua Library Trust había creado algo realmente bueno y se lo había regalado al mundo, y ahora Liblime quiere arrebatarlo” (Horowhenua Library Trust, 2011).

Actualmente se pueden encontrar dos ramas distintas de Koha, ambas de código abierto: desde noviembre de 2011 existe LibLime Koha (se encuentra en su versión 4.18) y por otro lado Koha Community (se encuentra en su versión 18.11.09). En este trabajo se ha decidido utilizar la versión de Koha Community.



Ilustración 14: Mapa de instalaciones de Koha. Fuente: Library Technology

Koha es una aplicación multiplataforma y se basa en una arquitectura cliente-servidor, cualquier navegador sirve como aplicación cliente ya que la interfaz, tanto para el usuario (OPAC) como para el bibliotecario (Intranet), son de tipo web. En cuanto al servidor, requiere un equipo con sistema operativo GNU/Linux o MS Windows, servidor Apache, lenguaje de programación Perl y sistema gestor de base de datos MySQL.

Además, Arriola Navarrete (2011) señala una serie de características de Koha a tener en cuenta:

- Compatible con MARC21 y UNIMARC
- Las plantillas en las que se basan las interfaces web están construidas con XHTML y CSS, superando en ambos casos las respectivas pruebas de validación
- Soporte multilingüe para las lenguas occidentales, orientales y escritura de derecha a izquierda
- Posibilidad de utilizar Zebra, motor de base de datos de alta velocidad que permite manejar grandes cantidades de información estructurada y en una variedad de formatos de entrada (correo electrónico, XML, MARC, etc.)

- Interfaz de programación de aplicaciones (API) que permite utilizar herramientas como los protocolos SRU/SRW¹⁷, Z39.50, herramientas como UnAPI¹⁸ y COinS/OpenURL¹⁹, etc.

En cuanto a iniciativas basadas en Koha en castellano caben destacar que han existido dos de gran interés: por un lado, Koha-UNLP en Argentina y por otro, Koha-Kobli en España.

Koha-UNLP fue desarrollado por la Facultad de Informática de la Universidad Nacional de La Plata (Buenos Aires, Argentina) y el cuál finalizó en 2012 cuando el Centro Superior para el Procesamiento de la Información (CeSPI), dependiente del UNLP, decidió desarrollar un nuevo sistema integral a partir de Koha-UNLP llamado Meran, el cuál veremos más adelante.

Koha-Kobli fue desarrollado por el Grupo de Trabajo las Bibliotecas de la Administración General del Estado (BAGEs) a través de la Subdirección General de Coordinación Bibliotecaria del Ministerio de Cultura y Deporte. Dicho grupo realizó un grupo de estudio y evaluación de los sistemas integrados de código abierto disponibles en el mercado con el objetivo de automatizar y unificar la gestión de las 982 bibliotecas pertenecientes a BAGEs. Kobli fue el resultado de añadir a Koha nuevas funcionalidades sugeridas por parte del Grupo de Trabajo, las cuales fueron añadidas posteriormente al desarrollo del Koha original. Tras 8 años, las BAGEs decidieron poner fin al proyecto Koha-Kobli el cual concluyó con el inicio del año 2019 y, a su vez, cerraron de forma definitiva su página web (<http://kobli.bage.es>).

En la siguiente ilustración encontramos un mapa de instalaciones de Koha en España (no recoge ninguna instalación en ninguno de los dos archipiélagos). Señalar que, de todos los SIGBs analizados, tan solo se observan instalaciones del sistema Koha en España utilizando esta herramienta:

¹⁷ SRU es un protocolo estándar de búsqueda y recuperación de consultas mediante el uso de CQL (Commun Query Language), una sintaxis normalizada de consulta. Para más información véase: <http://bvpb.mcu.es/es/estaticos/contenido.cmd?pagina=estaticos/sru>

¹⁸ UnAPI es una pequeña API HTTP que cualquier aplicación web puede utilizar para copiar contenido de forma discreta en páginas HTML. Es utilizado por otras aplicaciones como Evergreen o WordPress.

¹⁹ ContextObjets in Spans (COinS) es un método que incluye los metadatos bibliográficos relevantes de una pulicación científica en el código HTML de una página web.



Ilustración 15: Mapa de instalaciones de Koha en España. Fuente: Library Technology

Aunque en esta ilustración se muestran algunos centros donde Koha ha sido instalado, esta no se encuentra del todo actualizada. En la web del proyecto Kobli, encontramos una tabla mucho más detallada con los centros donde se ha instalado, en qué fase de implantación se encuentra y el enlace al OPAC de cada biblioteca (véase anexo E).

a) Generalidades

Características	Descripción
Nombre de la aplicación	Koha
Versión	18.11.09 (Agosto, 2019)
País de origen	Nueva Zelanda
Idiomas que maneja	Multilingüe (árabe, armenio, euskera, chino, checo, danés, inglés, finlandés, francés, alemán, griego, hindú, Italia, noruego, persa, polaco, portugués, eslovaco, español, sueco, turco y vietnamita).
Creador	Katipo Communications Ltd.
Licencia	GNU GPL (General Public License)
Página web	https://koha-community.org/
Página de descarga	https://koha-community.org/download-koha/

Características	Descripción
Soporte	Base de datos de errores conocidos: http://koha-community.org/support/search-bugs/ Foros: http://koha-community.org/support/forums/ Listas de discusión (varias listas): http://koha-community.org/support/koha-mailing-lists/ Listado de empresas de soporte: http://koha-community.org/support/paid-support/ Wiki: http://wiki.koha-community.org/wiki/Main_Page

Tabla 15: Generalidades de Koha

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux, MS Windows
Servidor	Apache
Leng. de programación	Perl
SGBD	MySQL

Tabla 16: Requisitos de instalación de Koha. Fuente: elaboración propia

c) Funcionalidades

Funcionalidad	Descripción
Mód. de administración	Si
Mód. de adquisiciones	Si
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	Si
Mód. gestión usuarios	Si
Módulo de préstamo interbibliotecario	Si
Módulo de reportes/mensajes	Si
Módulo de reservas	Si

Tabla 17: Funcionalidades de Koha

3.5 Meran



Meran es un SIGB de software libre desarrollado por el Centro Superior para el Procesamiento de la Información (CeSPI) en la Universidad Nacional de La Plata (UNLP)

y que, al igual que Koha, significa “regalo u obsequio” pero en Mapuche, lengua proveniente del pueblo amerindio del mismo nombre que habita en Chile y Argentina. El nombre se relaciona con la filosofía en la que se basa el trabajo desarrollado en la UNLP: ofrecer a la sociedad un sistema de libre uso adaptable a cualquier tipo de biblioteca.

Su desarrollo comenzó en julio de 2009 cuando el Grupo de Desarrollo del CeSPI decidió desarrollar un nuevo sistema integrado basándose en la experiencia de uso de la comunidad de Koha-UNLP. Se continuó ofreciendo soporte a los usuarios de Koha-UNLP.

Durante 2010 se realizó la puesta en producción de Meran y Koha-UNLP en la Biblioteca de la Facultad de Ciencias Económicas de la UNLP.

En 2011, El Grupo de Desarrollo incorporaría importantes cambios tecnológicos a Meran, entre ellos, una nueva interfaz innovadora y adaptativa. Sería en ese mismo año cuando Meran se implementa de forma definitiva en la Biblioteca de la Facultad de Ciencias Económicas de la UNLP y se crea la Comunidad Meran, formada por los usuarios provenientes de Koha-UNLP y nuevos usuarios interesados en el sistema.

Durante 2012, Meran se incorpora a las diferentes bibliotecas que conforman la UNLP y en octubre se produce la liberación del código de Meran en su versión 0.9 como software libre.



Ilustración 16: Mapa de instalación de Meran. Fuente: Library Technology

Meran es utilizado por gran parte de las facultades de la Universidad Nacional de La Plata y otros organismos provinciales y nacionales (véase anexo F).

Se basa en una arquitectura cliente-servidor, la interfaz tanto para el usuario (OPAC) como para el bibliotecario (Intranet) es de tipo web así que es posible acceder a esta utilizando cualquier navegador web. En cuanto al servidor, utiliza una arquitectura de tipo LAMP: sistema operativo GNU/Linux, lenguaje de programación PHP, servidor Apache y sistema gestor de base de datos MySQL.

Meran puede utilizarse desde un navegador web de teléfono celular. Además, ha sido desarrollado para promover la interacción social con sus usuarios; permite difundir servicios y productos utilizando redes sociales como Facebook y Twitter, entre otras; consultar el catálogo, recomendar bibliografía, etc. A su vez, la biblioteca también puede publicar noticias, novedades y mostrar materiales destacados a través de las redes sociales.

a) Generalidades

Características	Descripción
Nombre de la aplicación	Meran
Versión	0.10.7 (Julio, 2017)
País de origen	Argentina
Idiomas que maneja	Multilingüe (Español e inglés)
Creador	CeSPI (Centro Superior para el Procesamiento de la Información)
Licencia	GNU GPL (General Public License)
Página web	https://cespi.unlp.edu.ar/meran
Página de descarga	https://github.com/Desarrollo-CeSPI/meran/tags
Soporte	Manual de usuario: https://cespi.unlp.edu.ar/manual_del_usuario_meran

Tabla 18: Generalidades de Meran

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux
Servidor	Apache
Lenguaje de programación	Perl
Sistema Gestor de Base de Datos	MySQL

Tabla 19: Requisitos de instalación de Meran

c) Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	—
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	—
Módulo de gestión de usuarios	Si
Módulo de reportes/mensajes	Si

Tabla 20: Funcionalidades de Meran

3.6 OpenBiblio



OpenBiblio fue desarrollado por un grupo de informáticos a cargo de Dave Stevens en Estados Unidos en el año 2002, utilizando lenguaje de programación PHP. El objetivo fue desarrollar

un programa fácil de utilizar, de instalar y con las prestaciones requeridas para cubrir las necesidades de pequeñas bibliotecas, específicamente escolares. El proyecto es apoyado, como muchos otros SIGBs, por Sourceforge un portal para el desarrollo de software libre.

Hans van der Weij es el encargado actual de OpenBiblio y comenzó a usar este alrededor de 2003 cuando trabajaba en una biblioteca escolar en una escuela secundaria en los Países Bajos. OpenBiblio es actualizado de forma irregular, el proyecto parecía estar completamente abandonado pues desde 2008 y su versión 0.6.1 no fue actualizado hasta 4 años después, en 2012 y la versión 0.7.1, la cual actualizó el programa para que pudiera utilizar las versiones más recientes de MySQL y PHP. Su última versión estable es la 0.7.2, lanzada el 13 de agosto de 2014 y se centra principalmente en reforzar la seguridad del programa.

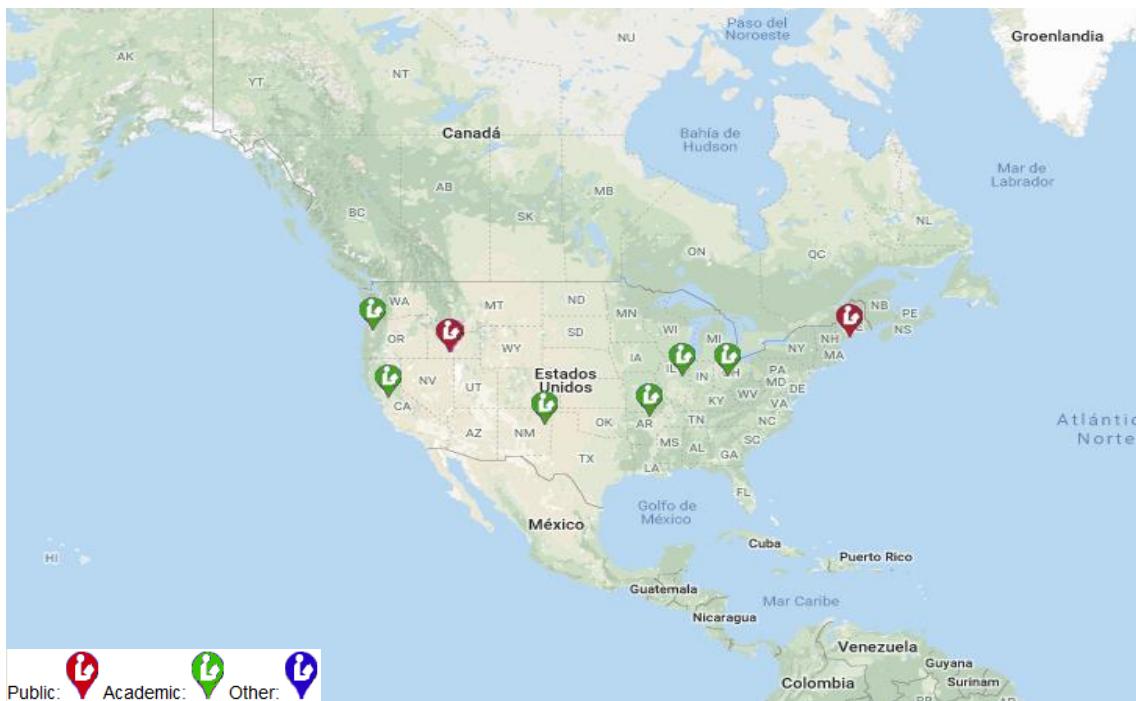


Ilustración 17: Mapa de instalaciones de OpenBiblio. Fuente: Library Technology

Funciona bajo una arquitectura cliente-servidor, sirviendo cualquier navegador web para la parte del cliente, pudiendo acceder tanto a la parte de usuarios (OPAC) como a la de bibliotecarios (Intranet). En cuanto al servidor, utiliza una arquitectura de tipo LAMP o WAMP: sistema operativo GNU/Linux o MS Windows, lenguaje de programación PHP, servidor Apache y base de datos MySQL.

Destacar que OpenBiblio solo permite importar en formato USMARC y no permite exportar registros en ningún formato, por lo tanto, sería necesario el uso de aplicaciones externas como MarcEdit (<https://marcedit.reeset.net/>) lo cual no es una tarea trivial.

Existe una iniciativa llamada EspaBiblio, una versión de OpenBiblio traducida al castellano y cuyo responsable es Jorge Lara Cravero. EspaBiblio está modificado para tener la capacidad de descargar materiales digitales y no solo es una traducción de OpenBiblio, sino que es un fork mejorado debido a que se integraron nuevas funciones, se corrigieron bugs y se actualizó para tener un mejor desempeño con las nuevas características de Apache, MySQL y PHP en sus últimas versiones. Sin embargo, su última actualización data de marzo de 2015.

Es utilizado en el sistema de educación primaria en Chile y países como Colombia, Cuba y Venezuela han expresado abiertamente su interés en este programa. La Biblioteca Nacional de Armenia también ha recomendado el uso de OpenBiblio para las más de 900 bibliotecas pequeñas y rurales del país. Además, Investigadores del Departamento de Ciencias de la información de la Universidad Federal de Paraíba (Brasil) han mostrado interés en el uso de este sistema para enseñar a los futuros bibliotecarios sobre los sistemas de automatización de bibliotecas²⁰.

²⁰ Para más información veáse: http://petermoulding.com/openbiblio_is_a_low_cost_software_option_for_libraries

OpenBiblio puede usar el directorio de menús y mensajes en castellano de EspaBiblio para dotar de este idioma a OpenBiblio, originalmente solo en inglés.

2.6 Generalidades

Características	Descripción
Nombre de la aplicación	OpenBiblio
Versión	0.7.2 ²¹ (Agosto, 2014)
País de origen	Estados Unidos
Idiomas que maneja	Inglés ²²
Creador	Dave Stevens
Licencia	GNU GPL (General Public License)
Página web	http://obiblio.sourceforge.net/
Página de descarga	v.0.7.2: https://sourceforge.net/projects/obiblio/files/latest/download Beta v1.0: https://bitbucket.org/mstetson/obiblio-1.0-wip/
Soporte	Wiki: http://obiblio.sourceforge.net/index.php/PmWiki/BasicEditing Foros: http://sourceforge.net/projects/obiblio/forums

Tabla 21: Generalidades de OpenBiblio

3.6 Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux y MS Windows
Servidor	Apache
Lenguaje de programación	PHP
Sistema Gestor de Base de Datos	MySQL

Tabla 22: Requisitos de instalación de OpenBiblio

²¹ La versión 1.0 a fecha de 8 de septiembre de 2019 se encuentra en fase beta por lo tanto se ha utilizado la versión en producción, es decir, la 0.7.2.

²² En <https://sourceforge.net/p/obiblio/patches/> hay disponibles otros idiomas, por ejemplo, italiano, ruso, polaco, holandés, etc. El español se puede obtener del fork EspaBiblio.

4.6 Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	—
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	—

Tabla 23: Funcionalidades de OpenBiblio

3.7 PMB



PMB, PhpMyBibli en sus primeras versiones, es un software de origen francés desarrollado por François Lemarchand, director de la Biblioteca de Agneaux, el cuál preparó las bases de catálogo y el esqueleto de la aplicación. El resto de funcionalidades se irían implementando con el paso del tiempo, como el módulo de publicaciones periódicas en 2002 o el OPAC en 2003, gracias a un equipo formado por Eric Robert, Gautier Michelin y Christophe Bliard, los cuales crearían la empresa PMB Services para profesionalizar el software y ofrecer soporte técnico para las instituciones interesadas.

François, el iniciador del proyecto, abandonaría el proyecto en 2003. El desarrollo de PMB comenzó en el año 2000, apareciendo la versión 1.0 en 2002 y la 1.2 en 2004. Actualmente se encuentra en su versión 5.0.7, liberada en marzo de 2019²³.

Es un SIGB potente y robusto, con una gran variedad de opciones de configuración, lo cual facilita su adaptación a diferentes tipos de organizaciones. Es perfectamente utilizable tanto en bibliotecas grandes o de pequeño tamaño, y lo suficientemente flexible como para adaptarlo a las necesidades de cada centro.

La primera biblioteca en ser equipada con PMB fue la biblioteca de Bueil-en-Touraine (Francia) y desde entonces ha sido implementada en más de 6.000 organizaciones de diversa índole, con colecciones que iban de 300 a 500.000 documentos. Grandes compañías como Alstom o incluso la empresa de telecomunicaciones francesa Orange S.A apuestan por el software gratuito como PMB para gestionar su documentación. Actualmente, PMB Services afirma tener más de 1.800 clientes y en el sitio web oficial de la compañía enumera la mayoría de ellos y enlaza sus catálogos en línea.

²³ Actualmente (septiembre 2019) existe una RC (Release Candidate) para PHP 7.x, la cual todavía no está en producción.



Ilustración 18: Mapa de instalaciones de PMB. Fuente: Library Technology

Inicialmente se distribuía bajo licencia GPL, aunque actualmente se distribuye bajo licencia CeCILL: una licencia francesa de software libre y una adaptación de la licencia General Public License (GPL) a la legislación francesa y otros países con sistemas legales similares.

PMB también utiliza arquitectura cliente-servidor, sirviendo cualquier navegador web en la parte del cliente pues la interfaz del programa es de tipo web, tanto para usuarios (OPAC) como para bibliotecarios (Intranet). En cuanto a la parte del servidor, utiliza una arquitectura de tipo LAMP o WAMP: sistema operativo GNU/Linux o MS Windows, lenguaje de programación PHP, servidor Apache y base de datos MySQL.

Incorpora el estándar de codificación de caracteres UNICODE, lo que hace de PMB un sistema de capacidad multilingüe no solo en la interfaz sino en los datos que maneja. Además, permite la consulta y recuperación de registros bibliográficos de otras bibliotecas a través del protocolo Z39.50.

a) Generalidades

Características	Descripción
Nombre de la aplicación	PHPMyBibli
Versión	5.0.7 (Marzo, 2019)
País de origen	Francia
Idiomas que maneja	Multilingüe (inglés, francés, español, italiano, árabe, danés y portugués).
Creador	François Lemarchand

Características	Descripción
Licencia	CeCILL (CEA CNRS INRIA Logiciel Libre) ²⁴
Página web	https://www.sigb.net/
Página de descarga	http://forge.sigb.net/redmine/projects/pmb/files
Soporte	https://pmb.community

Tabla 24: Generalidades de PMB

b) Requisitos de instalación

Requisito	Descripción
Sistema Operativo	GNU/Linux, MacOS y MS Windows
Servidor	Apache
Lenguaje de programación	PHP
Sistema Gestor de Base de Datos	MySQL

Tabla 25: Requisitos de instalación de PMB

c) Funcionalidades

Funcionalidad	Descripción
Módulo de administración	Si
Módulo de adquisiciones	—
Módulo de catalogación	Si
Módulo de circulación	Si
Módulo de estadísticas	Si
Módulo de OPAC	Si
Módulo de publicaciones periódicas	—
Módulo de D.S.I	Si
Módulo de gestión de usuarios	Si
Módulo de reportes/mensajes	Si

²⁴ <https://cecill.info/licences.fr.html>

Funcionalidad	Descripción
Módulo de reservas	Si

Tabla 26: Funcionalidades de PMB

4 COMPARACIÓN DE LOS SIGBS

a) Comparación de módulos

	ABCD	BIBLIO	Evergreen	Koha	Meran	OpenBiblio	PMB
Administración	Si	Si	Si	Si	Si	Si	Si
Adquisición	Si	–	Si	Si	–	–	–
Catalogación	Si	Si	Si	Si	Si	Si	Si
Circulación	Si	Si	Si	Si	Si	Si	Si
Difusión Selectiva de la Información	–	–	–	–	–	–	Si
Estadísticas	Si	Si	Si	Si	Si	Si	Si
Gestión de usuarios	Si	–	–	Si	Si	–	Si
OPAC	Si	–	Si	Si	Si	Si	Si
Préstamo interbibliotecario	–	–	–	Si	–	–	–
Publicaciones periódicas	Si	–	Si	Si	–	–	Si
Reservas	Si	–	–	Si	–	–	Si
Reportes/mensajes	–	–	–	Si	Si	–	Si

Tabla 27: Comparación de módulos de los SIGBs

b) Comparación de requerimientos

	Sistema Operativo	Servidor	Lenguaje de programación	SGBD
ABCD	GNU/Linux, MacOS MS Windows.	Apache	PHP, Java	MySQL
BIBLIO	GNU/Linux MS Windows	–	VisualBasic	Jet (MS Access)

	Sistema Operativo	Servidor	Lenguaje de programación	SGBD
Evergreen	GNU/Linux MacOS MS Windows.	Apache	Perl, C	PostgreSQL
Koha	GNU/Linux MS Windows	Apache	Perl	MySQL
Meran	GNU/Linux	Apache	Perl	MySQL
OpenBiblio	GNU/Linux MS Windows	Apache	PHP	MySQL
PMB	GNU/Linux MacOS MS Windows	Apache	PHP	MySQL

Tabla 28: Comparación de requerimientos de los SIGBs

c) Estándares para el intercambio de información bibliográfica

	Estándares
ABCD	Compatible con MARC21 y UNIMARC Compatible con Z39.50
BIBLIO	–
Evergreen	Compatible con la búsqueda y recuperación a través de direcciones URL y de servidores Z39.50
Koha	Compatible con MARC21 y UNIMARC Importación y exportación de registro bibliográficos Permite hacer reservas de fondos de otras sucursales
Meran	Compatible con MARC21
OpenBiblio	Compatible con USMARC (únicamente importación)
PMB	Compatible con UNIMARC Capacidad para importar registros bibliográficos completos

Tabla 29: Estándares para el intercambio de información bibliográfica

5 CONCLUSIONES

El objetivo principal del presente trabajo es ofrecer una panorámica de los SIGBs de software libre disponible en la actualidad para servir de orientación, principalmente, en dos entornos: el educativo y el profesional.

En entornos educativos, este estudio pretende ser de ayuda para estudiantes de asignaturas como “Sistemas Integrados para Unidades de Información”, cursada en nuestro Grado.

En entornos profesionales, este estudio puede ayudar a bibliotecarios con un problema que se encontrarán en varios momentos de su vida laboral: la elección e implementación de un nuevo SIGB por primera vez en su Biblioteca, o la migración de un SIGB anterior que se deba retirar por obsolescencia.

Para cumplir este objetivo se han usado las herramientas de búsqueda y elaborado el correspondiente listado recogiendo una gran cantidad de ellos con las herramientas disponibles.

El listado está conformado por aplicaciones tanto libres como gratuitas. Esto se debe a la escasa diferenciación que se aplica en Internet sobre el software libre y gratuito, por lo tanto, fue necesaria una posterior verificación para comprobar si realmente se trataban de sistemas de software libre mediante, por ejemplo, las licencias que los regulaban o si incluían el código fuente a la hora de descargarlos.

Además, se ha observado cierta disparidad a la hora de intentar definir el concepto de SIGB y los módulos que debe comprender para serlo, pues cada autor lo define de una manera distinta y cada aplicación está integrada por unos módulos diferentes, tanto en número como en la denominación de estos.

Hecho el estudio, se puede concluir que, en la actualidad hay un número limitado de SIGBs de software libre en comparación con la oferta de las aplicaciones comerciales. Dentro del software libre la diversidad disminuye, pues cada sistema está orientado a un tipo concreto de biblioteca, algunos ejemplos: OpenBiblio a bibliotecas de pequeño tamaño, Evergreen a grandes consorcios, Koha a bibliotecas de gran tamaño, etc.

Posteriormente, se utilizaron diversas plataformas para crear un entorno de aprendizaje (entornos educativos) y de pruebas (entornos profesionales) donde se pudieron instalar algunas aplicaciones. Podemos afirmar que, dependiendo del tipo de plataforma que utilicemos y la aplicación que queramos instalar, habrá una mayor o menor dificultad en el proceso.

AMPdoc está destinado a la educación, este paquete pone directamente a disposición OpenBiblio y PMB, entre otras aplicaciones del mundo de la Información y la Documentación como Omeka o Wordpress.

Las plataformas WAMP (especialmente XAMPP) incluyen un servidor Apache, un Sistema Gestor de Base de Datos (MySQL o MariaDB), y el lenguaje PHP. En estas plataformas es relativamente sencillo –y muy didáctico– la instalación de los citados OpenBiblio y PMB. Obviamente, esta plataforma solo sirve para pruebas, y –por motivos de seguridad– no debería usarse en sistemas reales de producción.

Las plataformas LAMP (varias implementaciones de GNU/Linux como Ubuntu) permiten instalar los citados OpenBiblio y PMB de una forma relativamente sencilla e incluso usarlos en sistemas reales de producción. Aplicaciones como Koha o Evergreen solo funcionarán en LAMP pero tras instalaciones más complejas. Ambas requieren el lenguaje Perl, Koha recomienda un indexador como Zebra y Evergreen usa el SGBD PostgreSQL, más potente que MySQL.

Por último, Koha y Evergreen pueden instalarse de una forma más sencilla mediante la virtualización de los mismos con VMware Player (gratuito) exclusivamente para su estudio y no en sistemas de producción.

Otro de los objetivos de este trabajo era poner a disposición una serie de características y funcionalidades de las aplicaciones instaladas. Vista la matriz comparativa propondríamos el siguiente orden de estudio y/o instalación experimental de las siguientes aplicaciones según un creciente grado de complejidad en todos los sentidos.

- Openbiblio (y otras igualmente sencillas como ABCD y BIBLIO): como primer ejemplo en formación, tanto de instalación como de uso, así como para bibliotecas de pequeño tamaño. Debería destacarse que con éstas sus actualizaciones son escasas y sus catálogos son difícilmente exportables.
- PMB: solución recomendable para educación por su relativamente sencilla instalación en varias plataformas (WAMP y LAMP) y por ser muy didáctica. Por otro lado, en el entorno profesional destaca por su robustez y adecuación en un amplio espectro de bibliotecas no excesivamente grandes y complejas. Muy interesantes resultan su política de actualizaciones y la larga lista de formatos bibliográficos MARC en los que se pueden importar/exportar sus registros.
- Koha (Merán) y Evergreen: instalaciones complejas debido a sus requerimientos para instalaciones en producción en entornos LAMP. Para formación podrá utilizarse versiones virtualizadas. Obviamente, para las bibliotecas grandes y complejas, o para consorcios (Evergreen) deberá utilizarse un entorno LAMP bajo la supervisión de personal informático, telemático y de seguridad de la institución, empresa o externo.
- Folio: todavía en desarrollo, pero habrá que seguir su evolución, ya que podría reemplazar a un SIGB muy extendido como Koha.

Como conclusión final afirmamos que la elección de un SIGB de software libre puede ser una opción arriesgada, pues nos encontramos ante entornos que se sustentan en las comunidades de usuarios. Es de vital importancia conocer y estudiar el sistema que queremos implantar, siendo este trabajo una guía. Además, deberán estudiarse sus comunidades de usuarios, y la existencia (o no) de empresas de soporte, que pueden ser indispensables.

Apostar por un sistema libre nos aporta varias ventajas en comparación al modelo propietario: mayor rapidez a la hora de obtener actualizaciones y mejoras, que dependen de la colaboración de usuarios cuya principal motivación es el correcto desarrollo de la aplicación, contrariamente al modelo propietario, cuya motivación principal es el beneficio económico.

Otro de los beneficios que nos aportaría el modelo libre sería el control total sobre nuestros datos, ya que disponemos de la totalidad de los códigos fuente y acceso completo a las Bases de Datos de todos nuestros registros. Esto rompe la dependencia con la empresa de soporte a diferencia de lo que sucede en el modelo propietario. Y proporciona un grado de libertad desconocido por las bibliotecas que han escogido SIGBs propietarios.

6 BIBLIOGRAFÍA

1. Agenjo Bullón, X. (1996). Directrices para la elección y/o desarrollo de un sistema integrado de gestión de bibliotecas. *Tratado básico de biblioteconomía*. 83-101. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=1033892>
2. Arriola Navarrete, O. (2011). Open Access y software libre: un área de oportunidad para las bibliotecas. *Biblioteca universitaria*. 14(1), 26-40. Recuperado de: <http://hdl.handle.net/10760/16172>
3. Arriola Navarrete, O., Tecuatl Quechol, G., y González Herrera, G. (2011). Software propietario vs software libre: una evaluación de sistemas integrales para la automatización de bibliotecas. *Investigación bibliotecológica*. 25(54), 37-70. Recuperado de: <http://hdl.handle.net/10760/16132>
4. Arriola Navarrete, O., y Butrón Yáñez, K. (2008). Sistemas integrales para la automatización de bibliotecas basados en software libre. *Acimed*. 18(6), 9. Recuperado de: <http://hdl.handle.net/10760/12760>
5. Arriola Navarrete, O., y Montes de Oca Aguilar, E. (2014). Sistemas Integrales de Automatización de Bibliotecas: una descripción suscita. *Bibliotecas y Archivos*. 1(3), 47-76. Recuperado de: <http://hdl.handle.net/10760/24259>
6. Benítez, J. (2015). *Mini Guía Básica de Servidores*. Recuperado de: <http://www.tecnopedia.net/wp-content/uploads/2015/07/Mini-Guia-B%C3%A1sica-Servidores-WAMP-LAMP-MAMP1.pdf>
7. Bilal, D. (2014). *Library Automation: Core Concepts and Practical Systems Analysis: Core Concepts and Practical Systems Analysis*. Santa Bárbara: Libraries Unlimited.
8. Blázquez, M. (2015, abril, 28). AMPdoc 2.0 (web log post). Recuperado de: <https://mblazquez.es/ampdoc-2-0/>
9. Bottcher, E. (2018, marzo, 5). What I Talk About When I Talk About Platforms (web log post). Recuperado de: <https://martinfowler.com/articles/talk-about-platforms.html#WhatIsAplatformAnyway>
10. Breeding, M. (2009). Major open source ILS products. *Library Technology Reports*. 44(8), 16-31. Recuperado de: <http://dx.doi.org/10.5860/ltr.44n8>
11. Breeding, M. (2009). Opening up library automation software. *Computers in Libraries*. 26(3). Recuperado de: <https://librarytechnology.org/repository/item.pl?id=13803>
12. Breeding, M. (2010). LibLime acquisition by PTFS marks a new era for Koha. *Library Journal*. 135(2). Recuperado de: <https://librarytechnology.org/document/14576>

13. Breeding, M. (2015). Library services platforms: A maturing genre of products. *Library Technology Reports*. 51(4). Recuperado de: <http://dx.doi.org/10.5860/ltr.51n4>
14. Breeding, M. (2017). Informe de sistemas de bibliotecas 2017. Visiones competitivas de tecnología, software abierto y flujo de trabajo. *El profesional de la información*. 21(9), 9-15. Recuperado de: <http://dx.doi.org/10.3145/epi.2012.ene.02>
15. Buonocore, D. (1976). *Diccionario de bibliotecología: términos relativos a la bibliología, bibliografía, bibliofilia, biblioteconomía, archivología, documentología, tipografía y materias afines*. Buenos Aires: Marymar.
16. Casanova Huerta, A. (2010). *Software libre una opción viable en la organización bibliográfica de fotografías: caso Biblioteca de las Artes* (Tesis doctoral). Escuela Nacional de Biblioteconomía y Archivonomía, Ciudad de México, México. Recuperado de: <http://hdl.handle.net/10760/7771>
17. CDS/ISIS. (3 de abril de 2019). En *Wikipedia, la enciclopedia libre*. Recuperado el 21 de junio de 2019 de <https://es.wikipedia.org/w/index.php?title=CDS/ISIS&oldid=115015258>
18. Centro Superior para el Procesamiento de la Información. (2013). *Manual Meran: Sistema integrado de gestión de bibliotecas*. Recuperado de: https://cespi.unlp.edu.ar/manual_del_usuario_meran
19. Chinchilla-Arley, R. (2011). El software libre: Una alternativa para automatizar unidades de información. *Bibliotecas*. 29(1). Recuperado de: <http://hdl.handle.net/10760/19389>
20. Ciprés Ortega, J. D., Hernández Ascencio, I. E., y Loa Valdez, C. E. (2014). *Sala infantil “Nahúm Pérez Paz” de la Escuela de Nacional de Biblioteconomía y Archivonomía: proyecto de automatización* (Tesis doctoral). Escuela Nacional de Biblioteconomía y Archivonomía, México. Recuperado de: <http://hdl.handle.net/10760/22515>
21. Cuesta Escudero, M. J. (1982). Los bibliotecarios y la automatización de Bibliotecas. *Boletín de ANABAD*. 32(3), 269-292. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=964809>
22. De Moya Anegón, F. (1995). *Los sistemas integrados de gestión bibliotecaria: estructuras de datos y recuperación de información*. Madrid: Anabad.
23. De Smet, E. (2010). Some ISIS-Software History and Technical Background on the New FOSS Integrated Library System ABCD. *LIBER Quarterly the journal of European Research Libraries*. 19(3-4), 324-335. Recuperado de: <https://dspace.library.uu.nl/handle/1874/241581>
24. De Smet, E. (2011). Special Features of the Advanced Loans Module of the ABCD Integrated Library System. *Program: Electronic Library and Information Systems*. 45(3), 323-332. Recuperado de: <http://dx.doi.org/10.1108/00330331111151629>
25. Deharbe, H. (2012). *Impacto de la implementación del Sistema de Gestión Bibliotecaria “Open MarcoPolo” en bibliotecas de Argentina y América Latina entre los*

- años 2003 y 2008 (Tesis doctoral). Universidad Nacional del Litoral, Santa Fé, Argentina. Recuperado de: <http://hdl.handle.net/11185/611>
26. Fernández Morales, M., y Chinchilla Arley, R. (2013). Automatización de unidades de información: Matriz técnica para la evaluación de software libre. *Revista Interamericana de Bibliotecología*. 36(3), 207-219. Recuperado de: <http://hdl.handle.net/10760/21135>
27. Flores Vargas, V. F. (2011). *El software de código abierto: una alternativa para la gestión integral de la biblioteca* (Tesis doctoral). Escuela Nacional de Biblioteconomía y Archivonomía, Ciudad de México, México. Recuperado de: <http://hdl.handle.net/10760/15464>
28. Free Software Foundation. (2007). *GNU General Public License*. Recuperado de: <https://www.gnu.org/licenses/gpl-3.0.html>
29. Free Software Foundation. (2019). *¿Qué es el software libre?* Recuperado de: <https://www.gnu.org/philosophy/free-sw.html>
30. Free Software Foundation. (2019). *¿Qué es GNU?* Recuperado de: <https://www.gnu.org/>
31. García Melero, L. Á., y García Camarero, E. (1999). *Automatización de bibliotecas*. Madrid: Arco/Libros.
32. Gómez Vega, E., y Elba Martín, A. (2016). Sistemas Integrales de Gestión para Bibliotecas. *Informes Científicos-Técnicos UNPA*. 8(1), 82-105. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=5454192>
33. González Fernández-Villavicencio, N. (2009). La biblioteca expandida en código abierto. *Boletín de la Asociación Andaluza de Bibliotecarios*. 24(96), 11-35. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=3347383>
34. Graña, J. (2012). *Manual de acceso rápido: "BIBLIO"*. Recuperado de: <http://www.abcdatos.com/tutorial/gestion-bibliotecas-manual.html>
35. Grupo de Trabajo de Catálogo Colectivo. (2010). Informe de evaluación del sistema integrado de gestión de bibliotecas Koha para las Bibliotecas de la Administración General del Estado. Disponible en: <http://hdl.handle.net/10421/4711>
36. Kumar, V., y Abraham, T. (2009). Eight things you should know about open source integrated library systems. Recuperado de: <http://hdl.handle.net/10760/13858>
37. López Yepes, J. (2004). *Diccionario enciclopédico de Ciencias de la Documentación*. Madrid: Síntesis.
38. Lorenzo, G. (2009). *Automatización de una planta industrial* (Tesis Doctoral). Universidad de Alicante, Departamento de Física, Ingeniería de Sistemas y Teoría de la Señal, Alicante, España. Recuperado de: <http://hdl.handle.net/10045/10056>
39. Martín-Gavilán, C. (2008). Temas de Biblioteconomía: El formato MARC: variedades geográficas y de aplicación. Recuperado de: <http://hdl.handle.net/10760/14525>

40. Meran: Sistema Integrado de Gestión de Bibliotecas. (2015). *El camino recorrido*. Recuperado de: https://cespi.unlp.edu.ar/articulo/2013/6/14/el_camino_recorrido#
41. Molyneux, R. E. (2009). Evergreen in context. *Bulletin of the American Society for Information Science and Technology*. 35(2), 26-30. Recuperado de: <https://doi.org/10.1002/bult.2008.1720350208>
42. Moreiro González, J. A. (2011). Evaluación de software libre para la gestión de archivos administrativos. *El profesional de la información*. 20(2), 206-213. Recuperado de: <http://hdl.handle.net/10016/21647>
43. Müller, T. (2008). *Étude comparative des principaux SIGB libres*. En Premier congrès de l'Association Internationale Francophone des Bibliothécaires et Documentalistes et satellite IFLA, Montréal, Canada, August 3 to 6. Recuperado de: <http://hdl.handle.net/10760/15389>
44. Müller, T. (2011) How to choose a free and open source integrated library system. *OCLC Systems & Services: International digital library perspectives*. 27(1), 57-78. Recuperado de: <http://hdl.handle.net/10760/15387>
45. Open Source Initiative. (2007). *The Open Source Definition*. Recuperado de: <https://opensource.org/docs/osd>
46. OpenBiblio. (21 de abril de 2018). En *Wikipedia, la enciclopedia libre*. Recuperado el 23 junio de 2019 de <https://es.wikipedia.org/w/index.php?title=OpenBiblio&oldid=107190277>
47. Picazo Cháfer, S. (2015). Panorámica general del uso de Sistemas Integrados de Gestión Bibliotecaria en España. *Métodos de información*. 6(10), 43-56. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=5157058>
48. Porcel Iturrealde, M. L., y Rodríguez Mederos, M. (2005). Software libre: una alternativa para las bibliotecas. *Revista Cubana de Información en Ciencias de la Salud*. 13(6), 8. Disponible en: <https://dialnet.unirioja.es/servlet/articulo?codigo=1417193>
49. Senso, J. A. (2011). Automatización de bibliotecas con PMB. Recuperado de: <http://hdl.handle.net/10760/15335>
50. Sintes Marco, B. (2018). *Instalación y uso de XAMPP en Windows*. Recuperado de: <http://www.mclibre.org/consultar/php/otros/xampp-instalacion-windows.html>
51. Sintes Marco, B. (2019). *Qué es XAMPP*. Recuperado de: <http://www.mclibre.org/consultar/php/otros/xampp.html>
52. Stallman, R. (2009). Why open source misses the point of free software. *Communications of the ACM*. 52(6), 31-33. Recuperado de: <https://www2.cs.duke.edu/courses/common/compsci092/papers/open/stallman-open-free.pdf>
53. Tennant, R. (2007). *Library Software Manifesto*. Recuperado de: <http://teches-sence.info/manifesto/>

54. Tramullas, J. (2010, febrero, 17). Koha, Liblime y lo que no hay que hacer [web log post]. Recuperado de: <https://tramullas.com/koha-liblime-y-lo-que-no-hay-que-aceptar/>
55. Tramullas, J. (2011, marzo, 29). Evergreen, software libre para informatizar bibliotecas [web log post]. Recuperado de: <https://tramullas.com/evergreen-software-libre-para-informatizar-bibliotecas/>
56. Tramullas, J. (2015, febrero, 11). Más software libre para bibliotecas y archivos [web log post]. Recuperado de: <https://tramullas.com/mas-software-libre-para-bibliotecas-y-archivos/>
57. Tramullas, J. (octubre, 2005). *Software libre para gestión de recursos de información digital*. En II Encuentro Internacional sobre Conocimiento Libre dentro de la II Conferencia Internacional de Software Libre, Mérida, España. 346-356. Disponible en: <http://hdl.handle.net/10760/7560>
58. Ubieto Artur, A. P. (2014). Software libre para sistemas integrados de bibliotecas, archivos y otras unidades de información en el "Grado en Información y Documentación". *Estudios de información, documentación y archivos: homenaje a la profesora Pilar Gay Molins.* 313-324. Recuperado de: <https://dialnet.unirioja.es/servlet/articulo?codigo=4773834>
59. Vasupongayya, S. (2011). Open source library management system software: a review. *International Scholarly and Scientific Research & Innovation.* 5(5), 820-825. Recuperado de: <https://waset.org/publications/8208/open-source-library-management-system-software-a-review>
60. Vázquez Díaz, J. J. (2013). *Software libre Kobli: una propuesta de automatización en la biblioteca escolar “Ing. Víctor Loredo Ortega” de la escuela secundaria técnica nº16 Tomas Alva Edison.* (Tesis doctoral). Escuela Nacional de Biblioteconomía y Archivonomía, México. Recuperado de: <http://www.bibliotecaenba.sep.gob.mx/tesis/Biblio2013/046063.pdf>
61. Virtualización. (3 de septiembre de 2019). En *Wikipedia, la enciclopedia libre*. Recuperado el 23 de junio de 2019 de <https://es.wikipedia.org/wiki/Virtualizaci%C3%B3n>
62. Wilson, K. (2006). *Computers in libraries: an introduction for library technicians*. Nueva York: Haworth Information Press.
63. Zurita Sánchez, J. M. (2008). Software libre: una alternativa para la gestión de información en bibliotecas. *Revista Iberoamericana de Sistemas, Cibernetica e Informática.* 5(1), 7-13. Recuperado de: <http://hdl.handle.net/10760/13294>

7 ANEXOS

Anexo A: Matriz técnica para la evaluación de software libre: generalidades

Número parámetro	Nombre del parámetro	Descripción
1	Nombre de la aplicación	Se refiere al nombre de la aplicación
2	Versión	Se refiere a la versión de la aplicación (este es muy importante, ya que las aplicaciones se liberan para el uso público en una determinada fecha, y se identifican con un número de versión). Una misma aplicación pueda estar funcionando en diferentes organizaciones, pero con números de versiones diferentes, lo cual las convierte en aplicaciones distintas.
3	País de origen	Corresponde al país de origen de la aplicación.
4	Idiomas que maneja	Es referente a los idiomas de la aplicación.
5	Creador	Contiene el creador de la aplicación
6	Licencia	Es referente al tipo de licencia de la aplicación, entre las cuales tenemos: <ul style="list-style-type: none">- Common Public License v.1.0.- GNU General Public License v.2.0.- GNU General Public License v.3.0.- Eclipse Public License.- eCos License v.2.0- Sleepycat Software Product License.- Affero License v.1.0- Affero License v.2.0- OpenSSL License
7	Página web	Se tiene la dirección URL de la página Web de la aplicación
8	Página de descarga	Indica la dirección URL de su página de descarga
9	Documentación	Dice si existe documentación técnica
10	Interfaz	Es referente a la interfaz que presenta la aplicación, ya sea si es a través de un browser que se visualiza y opera, o si es a través de una interfaz propia.
11	Plataforma	Se refiere a la plataforma en que se encuentra desarrollado el software: 1 si es un sistema de archivos, 2 si es cliente/servidor, y 3 si es una aplicación WEB.

Número parámetro	Nombre del parámetro	Descripción
12	Lenguaje(s) de programación	Corresponde a los lenguajes de programación utilizados para su desarrollo.
13	Modificación del código fuente	Se refiere a si se permite la modificación del código fuente.
14	Costo del software	Trata sobre el costo. Aunque se trata de software libre y no debería en principio tener costo, se verifica la existencia o no de algún costo por obtener la aplicación.
15	Mantenimiento anual	Va relacionado con el anterior, pero aquí se verifica la existencia o no de algún costo de mantenimiento anual
16	Actualización periódica	Es referente a la actualización periódica, con el fin de reflejar si el software está en revisión y actualización periódica o si por el contrario no ofrece la posibilidad de actualizaciones.
17	Motor de bases de datos	Incluye los motores de bases de datos soportados.
18	Robustez del motor de base de datos	El parámetro 19 está estrechamente relacionado con el parámetro 18, y se relaciona con la robustez del motor de base.
19	Soporte multi-biblioteca	Relacionado con la aplicación, ofrece la posibilidad de soportar múltiples bibliotecas. Este parámetro es importante ya que permite determinar si es posible el manejo de varias unidades de información de manera independiente con una sola instalación del software en un servidor.
20	Sistema Operativo	Está relacionado con los sistemas operativos en los cuales puede ejecutarse la aplicación.

Tabla 30: Generalidades del SIGB. Fuente: de Ricardo Chinchilla y Mynor Fernández

Anexo B: Matriz técnica para la evaluación de software libre: funcionalidades

Número parámetro	Nombre del parámetro	Descripción
1	Módulo Administrador	Se refiere a si la aplicación tiene un módulo de administración. El módulo Administrador es el que se encarga de todas las funciones administrativas y de configuración y parametrización de la aplicación.
2	Selección y Adquisiciones	Se refiere al módulo de Selección y Adquisiciones, responsable por facilitar y registrar los procesos de intercambios, compras y donaciones en una unidad de información.
3	Módulo Catalogación	Es referente con el módulo de Catalogación, que incluye las funciones relativas a los procesos técnicos relacionados con el desarrollo y mantenimiento de los catálogos bibliográficos.
4	Gestión de Autoridades	Es relativo a la Gestión de Autoridades, las cuales facilitan la utilización de una forma normalizada de los puntos de acceso de los catálogos automatizados, y muestra además las relaciones entre los distintos puntos de acceso.
5	Módulo de Circulación	Se refiere al módulo de Circulación, el cual es el responsable de todos los procesos relativos al préstamo y devoluciones de materiales por parte de los usuarios
6	Gestión de Publicaciones Periódicas	Se relaciona con la Gestión de Publicaciones Periódicas y el control de analíticas.
7	Catálogo Público en Línea (OPAC)	Es sobre si se tiene un Catálogo Público en línea, llamado también OPAC, para la consulta de información referencial por los usuarios desde sitios remotos a través de la Web
8	Módulo Estadísticas	Es sobre el módulo de Estadísticas, donde se refleja el uso de los distintos servicios ofrecidos en la unidad de información, lo cual permite hacer estudios cuantitativos de usabilidad con miras a impulsar acciones para su mejoramiento
9	Gestión de Tesauro	Es referente a la Gestión de Tesauros, los cuales permiten una normalización terminológica para mejorar el canal de acceso y comunicación entre los usuarios y la unidad de información.
10	Módulo de seguridad	Se refiere al módulo de Seguridad, que es el responsable de los procesos relativos con la protección y confiabilidad de la información que se administra.
11	Módulo de Inventario	Es el responsable de la gestión y mantenimiento del inventario de materiales de la unidad de información

Número parámetro	Nombre del parámetro	Descripción
12	Servicio de Información y Referencia	Se refiere al Servicio de Información y Referencia, módulo que gestiona y administra las solicitudes de información que realiza el usuario y que requieren de apoyo bibliotecológico directo para su atención.
13	Servicio de Difusión Selectiva de la Información	Se relaciona con los servicios de Difusión Selectiva de la Información que se ofrecen a los usuarios para atender sus necesidades programadas de información.
14	Indización y Resumen	Se enfoca en el módulo de Indización y Resumen, el cual facilita el análisis documental de documentos digitales para facilitar su acceso
15	Signaturización y Etiquetado	Se relaciona con la Signaturización y Etiquetado, para la generación de firmas y etiquetas de código de barras para identificar la colección de la unidad de información y facilitar los procesos de préstamo y devolución.
16	Variedad formatos visualización	Se relaciona con la Variedad de formatos que presenta el software para visualización de la información, y qué tan flexible es la aplicación para cambiar dichos formatos.

Tabla 31: Funcionalidades del SIGB. Fuente: de Ricardo Chinchilla y Mynor Fernández

Anexo C: Derechos y responsabilidades del consumidor

Derechos como consumidor
<ul style="list-style-type: none">- Tengo derecho a saber lo que existe ahora y cuál es su potencial funcionalidad futura.- Tengo derecho a usar lo que compro.- Tengo derecho al API si he comprado el producto.- Tengo derecho a documentación completa y actualizada.- Tengo derecho a mis datos.- Tengo derecho a tener acceso de sólo lectura a la base de datos.- Tengo derecho a no hacer las cosas sencillas innecesariamente complicadas.- Tengo derecho a conocer las líneas de desarrollo y la estimación de tiempo de desarrollo del producto que he comprado.- Tengo derecho a hacer preguntas técnicas a un equipo capaz de comprenderlas y responderlas.- Tengo derecho a no ser un probador involuntario.- Tengo derecho a que se conserven mis personalizaciones y configuraciones en futuras actualizaciones.
Responsabilidades como consumidor
<ul style="list-style-type: none">- Tengo la responsabilidad de conocer las necesidades de mis usuarios.- Tengo la responsabilidad de poner las necesidades de mis usuarios por delante de las mías.- Tengo la responsabilidad de comunicar mis necesidades clara y específicamente.- Tengo la responsabilidad de que las mejoras que pido sean realmente lo que quiero.- Tengo la responsabilidad de asignar honestamente las prioridades de las mejoras.- Tengo la responsabilidad de darme cuenta de que no soy especial.- Tengo la responsabilidad de elegir software usando un procedimiento limpio y razonable.- Tengo la responsabilidad de informar de los errores reproducibles de forma que puedan reproducirse.- Tengo la responsabilidad de informar de los errores irreproducibles con todos los detalles que pueda.- Tengo la responsabilidad de ver críticamente cualquier ajuste a las configuraciones predefinidas.
Responsabilidades compartidas
<ul style="list-style-type: none">- Tenemos la responsabilidad de comenzar desde una posición de respeto mutuo.- Tenemos la responsabilidad de comunicarnos correctamente.- Tenemos la responsabilidad de establecer y mantener un proceso de mejora racional.

Derechos como consumidor
<ul style="list-style-type: none">- Tenemos la responsabilidad de mantener las necesidades del usuario final como primordiales.- Tenemos la responsabilidad de relajarnos y divertirnos

Tabla 32: Derechos y responsabilidades del consumidor. Fuente: Roy Tennant

Anexo D: Licencias de software libre existentes en la actualidad

License	Author	Latest version	Publi-cation date	Linking	Distribution	Modification	Patent grant	Private use	Sublicensing	TM grant
Academic Free License^[10]	Lawrence E. Rosen	3.0	2002	Permissive	Permissive	Permissive	Yes	Yes	Permissive	No
Affero General Public License	Affero Inc	2.0	2007	Copylefted ^[11]	Copyleft except for the GNU AGPL^[11]	Copyleft ^[11]	?	Yes ^[11]	?	?
Apache License	Apache Software Foundation	2.0	2004	Permissive ^[12]	Permissive ^[12]	Permissive ^[12]	Yes ^[12]	Yes ^[12]	Permissive ^[12]	No ^[12]
Apple Public Source License	Apple Computer	2.0	August 6, 2003	Permissive	?	Limited	?	?	?	?
Artistic License	Larry Wall	2.0	2000	With re-strictions	With restrictions	With restrictions	No	Permis-sive	With restrictions	No
Beerware	Poul-Henning Kamp	42	1987	Permissive	Permissive	Permissive	No	Permis-sive	Permissive	No
BSD License	Regents of the University of California	3.0	?	Permissive ^[13]	Permissive ^[13]	Permissive ^[13]	Manu-ally ^[13]	Yes ^[13]	Permissive ^[13]	Manu-ally ^[13]
Boost Software License	?	1.0	August 17, 2003	Permissive	?	Permissive	?	?	?	?

License	Author	Latest version	Publication date	Linking	Distribution	Modification	Patent grant	Private use	Sublicensing	TM grant
Creative Commons Zero	Creative Commons	1.0	2009	Public Domain ^{[14][15]}	Public Domain	Public Domain	No	Public Domain	Public Domain	No
CC-BY	Creative Commons	4.0	2002	Permissive ^[16]	Permissive	Permissive	No	Yes	Permissive	?
CC-BY-SA	Creative Commons	4.0	2002	Copylefted ^[16]	Copylefted	Copylefted	No	Yes	No	?
CeCILL	CEA / CNRS / INRIA	2.1	June 21, 2013	Permissive	Permissive	Permissive	No	Permissive	With restrictions	No
Common Development and Distribution License	Sun Microsystems	1.0	December 1, 2004	Permissive	?	Limited	?	?	?	?
Common Public License	IBM	1.0	May 2001	Permissive	?	Copylefted	?	?	?	?
Cryptix General License	Cryptix Foundation	N/A	1995	Permissive	Permissive	Permissive	Manually	Yes	?	Manually
Eclipse Public License	Eclipse Foundation	2.0	August 24, 2017	Limited ^[17]	Limited ^[17]	Limited ^[17]	Yes ^[17]	Yes ^[17]	Limited ^[17]	Manually ^[17]
Educational Community License	Indiana University^[18]	1.0	2007	Permissive	?	Permissive	?	?	?	?

License	Author	Latest version	Publication date	Linking	Distribution	Modification	Patent grant	Private use	Sublicensing	TM grant
European Union Public Licence	European Commission	1.2	May 2017	Copylefted, with an explicit compatibility list ^[19]	Copylefted, with an explicit compatibility list ^[19]	Copylefted, with an explicit compatibility list ^[19]	Yes ^[20]	Yes ^[20]	Copylefted, with an explicit compatibility list ^[19]	No ^[20]
GNU Affero General Public License	Free Software Foundation	3.0	2007	GNU GPLv3 only ^[21]	Copylefted ^[22]	Copylefted ^[22]	Yes ^[23]	No network usage ^[23]	Copylefted ^[22]	Yes ^[23]
GNU General Public License	Free Software Foundation	3.0	June 2007	GPLv3 compatible only ^{[24][25]}	Copylefted ^[22]	Copylefted ^[22]	Yes ^[26]	Yes ^[26]	Copylefted ^[22]	Yes ^[26]
GNU Lesser General Public License	Free Software Foundation	3.0	June 2007	With restrictions ^[27]	Copylefted ^[22]	Copylefted ^[22]	Yes ^[28]	Yes	Copylefted ^[22]	Yes ^[28]
IBM Public License	IBM	1.0	August 1999	Copylefted	?	Copylefted	?	?	?	?
ISC license	Internet Systems Consortium	N/A	June 2003	Permissive	Permissive	Permissive	?	?	?	?
LaTeX Project Public License	LaTeX project	1.3c	?	Permissive	?	Permissive	?	?	?	?
Microsoft Public License	Microsoft	N/A	?	Permissive	Permissive	Permissive	No	Permissive	?	No

License	Author	Latest version	Publication date	Linking	Distribution	Modification	Patent grant	Private use	Sublicensing	TM grant
MIT license / X11 license	MIT	N/A	1988	Permissive ^[29]	Permissive ^[29]	Permissive ^[29]	Manually ^[29]	Yes ^[29]	Permissive ^[29]	Manually ^[29]
Mozilla Public License	Mozilla Foundation	2.0	January 3, 2012	Permissive ^[30]	Copylefted ^[30]	Copylefted ^[30]	Yes ^[30]	Yes ^[30]	Copylefted ^[30]	No ^[30]
Netscape Public License	Netscape	1.1	?	Limited	?	Limited	?	?	?	?
Open Software License^[10]	Lawrence Rosen	3.0	2005	Permissive	Copylefted	Copylefted	Yes	Yes	Copylefted	?
OpenSSL license	OpenSSL Project	N/A	?	Permissive	?	Permissive	?	?	?	?
Python Software Foundation License	Python Software Foundation	2	?	Permissive	?	Permissive	?	?	?	?
Q Public License	Trolltech	?	?	Limited	?	Limited	?	?	?	?
Sleepycat License	Sleepycat Software	N/A	1996	Permissive	With restrictions	Permissive	No	Yes	No	No
Unlicense	unlicense.org	1	December 2010	Permissive/Public domain	Permissive/Public domain	Permissive/Public domain	?	Permissive/Public domain	Permissive/Public domain	?
W3C Software Notice and License	W3C	20021231	December 31, 2002	Permissive	?	Permissive	?	?	?	?

License	Author	Latest version	Publication date	Linking	Distribution	Modification	Patent grant	Private use	Sublicensing	TM grant
Do What The Fuck You Want To Public License (WTFPL)	Banlu Kemiya-torn, Sam Hocevar	2	Decem-ber 2004	Permis-sive/Public domain	Permissive/Public domain	Permissive/Public domain	No	Yes	Yes	No
XCore Open Source License also separate "Hardware License Agreement"	XMOS	?	Febru-ary 2011	Permissive	Permissive	Permissive	Manu-ally	Yes	Permissive	?
XFree86 1.1 License	The XFree86 Project, Inc	?	?	Permissive	?	Permissive	?	?	?	?
zlib/libpng license	Jean-Loup Gailly and Mark Adler	?	?	Permissive	?	Permissive	?	?	?	?

Tabla 33: Licencias de software libre. Fuente: [Wikipedia](#)

Anexo E: Instalaciones de Koha en España y Latinoamérica

Biblioteca	Fase	OPAC
Biblioteca 2.0 del Instituto Minero de España en Zaragoza	<i>Producción</i>	OPAC
Biblioteca Antonio Martínez de Castro (Méjico)	<i>Producción</i>	OPAC
Biblioteca Central del Ministerio de Presidencia	<i>Producción</i>	OPAC
Biblioteca de Cultura	<i>Producción</i>	OPAC
Biblioteca de l'Arxiu Històric de CCOO de Catalunya (AHCO)	<i>Producción</i>	OPAC
Biblioteca de la Comisión Nacional de los Mercados y la Competencia – Competencia	<i>Producción</i>	OPAC
Biblioteca de la Fundació Bonanova	<i>Producción</i>	OPAC
Biblioteca de la Fundación Internacional Baltasar Garzón	<i>Producción</i>	OPAC
Biblioteca de Las Naves. Espacio de creación contemporánea	<i>Producción</i>	OPAC
Biblioteca del Colegio San Ignacio de Loyola	<i>Producción</i>	OPAC
Biblioteca del Consejo Económico y Social	<i>Producción</i>	OPAC
Biblioteca del Consejo Superior de Deportes	<i>Producción</i>	OPAC
Biblioteca del Hospital Universitario de Fuenlabrada	<i>Implantación</i>	–
Biblioteca del I.E.S. Jaime Vera. Madrid	<i>Implantación</i>	–
Biblioteca municipal de El Catllar. Tarragona	<i>Producción</i>	OPAC

Biblioteca	Fase	OPAC
Biblioteca pública municipal de Becerril de la Sierra	<i>Producción</i>	OPAC
Biblioteca Universitaria Miguel de Cervantes (El Salvador)	<i>Producción</i>	OPAC
Bibliotecas del Centro de Análisis y Prospectiva y Academia de Oficiales de la Guardia Civil	<i>Producción</i>	OPAC
Centro de Documentación de Música y Danza del Instituto Nacional de las Artes Escénicas y de la Música (INAEM)	<i>Producción</i>	OPAC
Centro de Documentación de Propiedad Intelectual	<i>Producción</i>	OPAC
Centro de Documentación del Libro, la Lectura y las Letras	<i>Producción</i>	OPAC
Centro de Documentación del Transporte del Ministerio de Fomento	<i>Producción</i>	OPAC
CIDE – Corporación Internacional para el Desarrollo Educativo. Bogotá (Colombia)	<i>Producción</i>	OPAC
Consejo Nacional de Investigaciones Científicas y Técnicas. Catálogo Colectivo de Libros (Conicet-CCLC). Buenos Aires (Argentina)	<i>Producción</i>	OPAC
Consorcio de Bibliotecas Universitarias de El Salvador. Catálogo de recursos electrónicos	<i>Producción</i>	OPAC
CRAI de la UMET. Centro de Recursos para el Aprendizaje e Investigación. Guayaquil (Ecuador)	<i>Producción</i>	OPAC
CRAI de la UMET. Centro de Recursos para el Aprendizaje e Investigación. Vozandes (Ecuador)	<i>Producción</i>	OPAC
HisTra – Historia de la traducción	<i>Producción</i>	OPAC
Servicio de Documentación de la Agencia Estatal del Boletín Oficial del Estado	<i>Implantación</i>	–

Tabla 34: Instalaciones de Koha. Fuente: [BAGEs](#)

Anexo F: Instalaciones de Meran**Bibliotecas de Facultades de la UNLP:**

- [Biblioteca Conjunta \(Facultad de Ciencias Agrarias y Forestales y Facultad de Ciencias Veterinarias\)](#)
- [Biblioteca de la Facultad de Arquitectura y Urbanismo](#)
- [Biblioteca de la Facultad de Bellas Artes](#)
- [Biblioteca de la Facultad de Ciencias Económicas](#)
- [Biblioteca de la Facultad de Ciencias Exactas](#)
- [Biblioteca de la Facultad de Ciencias Médicas](#)
- [Biblioteca de la Facultad de Informática](#)
- [Biblioteca de la Facultad de Psicología](#)
- [Biblioteca de la Facultad de Trabajo Social](#)

Colegios y dependencias de la UNLP:

- [Biblioteca de la Escuela Graduada Joaquín V. González](#)
- [Biblioteca del Bachillerato de Bellas Artes](#)
- [Biblioteca del Colegio Nacional Rafael Hernández](#)
- [Biblioteca del Museo Azzarini](#)
- [Biblioteca Pública](#)
- [Biblioteca Teatral](#)
- [Digesto UNLP](#)

Instituciones de derechos humanos y gubernamentales:

- [Biblioteca de la Comisión Provincial por la Memoria](#)
- [Biblioteca del Espacio Cultural Nuestros Hijos \(ECuNHi\)](#)
- Biblioteca Juan Benito Juárez (Municipalidad de Gral. Belgrano)

Institutos de Investigación:

- [Biblioteca del Instituto de Botánica Darwinion](#) (CONICET - ANCEFN)
- [Instituto de Historia del Arte Argentino y Americano \(IHAAA\)](#)