



Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Centro de Enseñanza Asistida por Computador - CENEAC

**Desarrollo del Módulo “Trabajos Académicos” del Portal  
*Biblioteca Virtual Musicológica Juan Meserón* de la  
Escuela de Artes, Universidad Central de Venezuela**

Trabajo Especial de Grado presentado ante la Ilustre  
Universidad Central de Venezuela por el Bachiller:

Víctor D. Cardozo Fernández C.I. 19.873.236

Para optar por el título de Licenciado en Computación

Tutora:

Profesora Yusneyi Y. Carballo Barrera

Caracas, 01 de Mayo de 2015

## **ACTA**

Quienes suscriben, miembros del jurado designado por el Consejo de la Escuela de Computación, para examinar el Trabajo Especial de Grado titulado “**Desarrollo del Módulo “Trabajos Académicos” del Portal *Biblioteca Virtual Musicológica Juan Meserón* de la Escuela de Artes, Universidad Central de Venezuela**”, presentado por el Br. **Víctor Daniel Cardozo Fernández (C.I. V-19.873.236)**, a los fines de optar al título de **Licenciado en Computación**, dejamos constancia de lo siguiente:

Leído como fue dicho trabajo, por cada uno de los miembros del jurado, se fijó el día \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_, a las \_\_\_\_\_ horas, para que el autor lo defendiera en forma pública, lo que este hizo en \_\_\_\_\_ de la Escuela de Computación, mediante una presentación oral de su contenido, luego de lo cual respondió a las preguntas formuladas. Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobar con la nota de \_\_\_\_ puntos.

En fe de lo cual se levanta la presente Acta, en Caracas el día \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_.

---

**Profa. Yusneyi Carballo Barrera**  
(Tutora)

---

**Profa, Mariantonia Palacios**  
(Jurado)

---

**Prof, Wilfredo Rangel**  
(Jurado)

## Agradecimientos y Dedicatorias

Mi profundo agradecimiento a los Profesores de la Escuela de Arte: Mariantonia Palacios por haberme ofrecido el tema de TEG y a mi Tutora Yusneyi Carballo Barrera por guiarme para un correcto desarrollo del mismo.

A todo el profesorado de la Escuela de Computación, por su esmero en otorgarme los conocimientos necesarios para convertirme en buen profesional.

A mi familia que siempre me dio todo el apoyo necesario para seguir adelante y culminar con éxito todas las metas.

A mis amigos por brindarme su amistad y su apoyo que nos hicieron vivir tantos buenos momentos.



Universidad Central de Venezuela

Facultad de Ciencias

Escuela de Computación

Centro de Enseñanza Asistida por Computador – CENEAC

**Desarrollo del Módulo “Trabajos Académicos” del Proyecto *Biblioteca Virtual Musicológica Juan Meserón* de la Escuela de Artes, Universidad Central de Venezuela**

**Autor:** Víctor Daniel Cardozo Fernández

**CI:** 19.873236

**Correo:** [bigtor.cardozo@gmail.com](mailto:bigtor.cardozo@gmail.com)

**Tutora:** Profa. Yusneyi Carballo Barrera

**Fecha:** 13 de abril de 2015

## **RESUMEN**

El presente Trabajo Especial de Grado se enfocó en el desarrollo del Módulo “Trabajos Académicos” en el marco del Proyecto Biblioteca Virtual Musicológica Juan Meserón de la Escuela de Artes de la Universidad Central de Venezuela. Este módulo está dirigido a todas las personas que tengan interés en consultar, leer y descargar Trabajos Especiales de Grado, Tesis de Postgrado, Trabajos de Ascenso, Monografías, entre otros documentos relacionadas con las investigaciones musicales generadas en el Departamento de Musicología de la Escuela de Artes. Para el desarrollo de este módulo se combinaron las tecnologías PHP (lenguaje de programación), CakePHP (*framework* o marco de trabajo), MySQL (Sistema Manejador de Base de Datos) y el proceso de desarrollo Scrum.

**Palabra Claves:** Biblioteca Virtual, Repositorio académico en línea, Trabajos Académicos, Catalogación en Formato MARC 21, PHP, CakePHP, MySQL.

# Índice

<b>INTRODUCCIÓN .....</b>	<b>7</b>
<b>CAPÍTULO I. Planteamiento del problema .....</b>	<b>8</b>
1.1. Análisis del Problema .....	8
1.2. Objetivos del Trabajo Especial de Grado.....	8
1.2.1. Objetivo General .....	8
1.2.2. Objetivos Específicos .....	8
1.2.3. Alcance de la Aplicación .....	9
1.3 Potenciales usuarios .....	9
<b>CAPÍTULO II. Marco Teórico .....</b>	<b>10</b>
2.1. Bibliotecas Virtuales .....	10
2.2. Marco Conceptual .....	10
2.2.1. Módulo .....	11
2.2.2. Trabajo Académico .....	11
2.2.3. Tesis .....	11
2.2.4. MARC 21 .....	12
2.2.5. Administrador de Contenido .....	13
2.2.6. Buscador.....	13
2.2.7. Base de Datos .....	14
2.2.8. Sistema Manejador de Base de Datos .....	14
<b>CAPÍTULO III. Marco Metodológico y Herramientas de Desarrollo .....</b>	<b>15</b>
3.1. Tipo de Investigación.....	15
3.2. Método de Desarrollo Ágil de Software.....	15
3.3. Metodología para el Desarrollo de la Aplicación .....	17
3.4. Tecnologías para el desarrollo de la aplicación. ....	21
3.4.1. PHP (PHP Hypertext Pre-processor).....	21
3.4.2. CakePHP .....	22
3.4.3. MySQL.....	26
3.4.4. Otras herramientas para el desarrollo .....	26
3.5 Implementación del Formato MARC 21 .....	28
3.5.1. Cabecera .....	28
3.5.2. Campos de Longitud Fija .....	30
3.5.3. Campos de Longitud Variable .....	32

3.5.4 Ejemplo.....	33
<b>CAPÍTULO IV. Marco Aplicativo .....</b>	<b>34</b>
4.1. Perfiles de usuarios .....	34
4.2. Requerimientos del módulo desarrollado y de la BVMJM .....	35
4.3. Requisitos del Sistema .....	35
4.3.1. Requisitos Funcionales .....	35
4.3.2. Requisitos no Funcionales .....	36
4.4. Definición y análisis del modelo de datos .....	36
4.5. Principales interfaces de la Biblioteca Virtual Musicológica “Juan Meserón” .....	38
4.5.1. Página principal de la Biblioteca Virtual.....	38
4.5.2. Interfaces de administrador.....	40
4.6. Aplicación de la metodología Scrum.....	47
4.7. Lista de tareas de las iteraciones .....	47
4.8. Pruebas.....	49
4.9. Análisis de Resultados .....	50
<b>Conclusiones .....</b>	<b>51</b>
<b>Recomendaciones y Trabajos Futuros .....</b>	<b>52</b>
<b>Referencias Bibliográficas y Digitales .....</b>	<b>53</b>
<b>Anexos.....</b>	<b>56</b>
Anexo A .....	56

## Índice de tablas

Tabla 1. Cuadro comparativo entre metodologías ágiles y tradicionales .....	16
Tabla 2. Lista de objetivos por sprint.....	47

## Índice de Figuras

Figura 1. Especificación de la Metodología Cascada. ....	18
Figura 2. Proceso del desarrollo de un proyecto utilizando Scrum .....	19
Figura 3. Típica petición en CakePHP usando MVC .....	25
Figura 4. Formato MARC 21 .....	33
Figura 5. Página principal de la Biblioteca Virtual Musicológica Juan Meserón .....	39
Figura 6. Ingreso de usuario .....	41
Figura 7. Página principal del módulo de administración.....	42
Figura 8. Página principal Módulo de Trabajos Académicos .....	44
Figura 9. Agregar Trabajo Académico.....	45
Figura 10. Detalle Trabajo Académico .....	46

# INTRODUCCIÓN

Dado el impacto y el alcance de las tecnologías de información en materia de comunicación y divulgación del conocimiento, la Escuela de Artes en la Facultad de Humanidades y Educación se ha propuesto, en colaboración con la Escuela de Computación de la Facultad de Ciencias, ambas de la Universidad Central de Venezuela, han cooperado en el desarrollo de la Biblioteca Virtual Musicológica Juan Meserón (BVMJM), un portal orientado a la exposición y preservación de la información asociada a libros, revistas, periódicos, partituras, reseñas, investigaciones sobre obras de carácter musical, etc., portal en el que se viene trabajando de forma modular.

La Escuela de Artes ha ido recopilando a lo largo de su historia una gran cantidad de trabajos académicos relacionados con la música de gran interés para el estudio de la música en Venezuela y Latinoamérica. Sin embargo, estos documentos no son de fácil acceso para todas las personas que pudieran estar interesadas en ese contenido. Motivado por eso, la Escuela de Artes se ha propuesto la tarea de promover y difundir ese material. Para poder difundir estos materiales, se propuso desarrollar el módulo “Trabajos Académicos” integrado al Portal BVMJM, que es el objeto de este Trabajo Especial de Grado.

Para dar cumplimiento a los requerimientos identificados, se plantea desarrollar una aplicación que combine las siguientes tecnologías: Esquema de catalogación de documentos usando el estándar MARC 21, PHP como lenguaje de programación en su versión 5.5.5, CakePHP como *framework* de desarrollo en su versión 2.4.3, MySQL como Sistema Manejador de Base de Datos en su versión 5.6.11, Apache 2.4.4 como servidor web y Scrum como metodología de desarrollo ágil.



# **CAPÍTULO I. Planteamiento del problema**

## **1.1. Análisis del Problema**

La Escuela de Artes de la Facultad de Humanidades y Educación resguarda una gran cantidad de material impreso y digital importante relacionado con la música, incluyendo los trabajos académicos desarrollados por estudiantes de pregrado y postgrado, así como trabajos de ascenso de profesores del Departamento de Música.

Para la preservación, exposición, consulta y difusión de estos trabajos académicos se presenta la necesidad de la creación de un módulo dentro de la BVMJM.

El desarrollo de la BVMJM se encuentra a cargo de varios estudiantes tesistas de la Escuela de Computación, quienes trabajan en la creación de los diferentes módulos del portal: el módulo hemerografía (Camargo, 2012), el módulo libros (Guzmán y Quintana, 2011), el módulo iconografía musical (Gamboa, 2014) y el módulo de partituras manuscritas e impresas (Gamboa, 2012). Se une a este proyecto el módulo “Trabajos Académicos”, el cual debe mantener la consistencia en cuanto a interfaz y diseño y debe integrarse a los otros módulos de la BVMJM

## **1.2. Objetivos del Trabajo Especial de Grado**

### **1.2.1. Objetivo General**

Desarrollar el Módulo “Trabajos Académicos” del Proyecto *Biblioteca Virtual Musicológica Juan Meserón*, integrando funcionalidades de catalogación, consulta, visualización y descarga de documentos académicos de contenido musical, a fin de promover la difusión y conocimiento de los trabajos generados en el Departamento de Música de la Escuela de Artes de la UCV.

### **1.2.2. Objetivos Específicos**

1. Emplear un proceso de desarrollo de software para la creación del módulo “Trabajos Académicos” del portal BVMJM que nos permita un avance organizado, dinámico, rápido y evolutivo del módulo.
2. Realizar el diseño de la base de datos que permita almacenar de manera correcta los Trabajos Académicos adecuadamente catalogados.

3. Aplicar el modelo del diseño web, estándares, patrones y lineamientos de usabilidad aplicados por los tesisistas que trabajaron en los módulos previos para lograr un diseño acorde con los otros módulos del portal.
4. Desarrollar un buscador para facilitar la búsqueda de Trabajos Académicos dentro del portal.
5. Evaluar el buen funcionamiento del módulo de Trabajos Académicos a través de la realización de pruebas de aceptación, para garantizar la satisfacción de los usuarios durante su navegación por el portal.
6. Diseñar la interfaz para el catalogador en la cual se despliegue la ficha catalográfica basada en el formato Marc21,
7. Documentar el desarrollo y el análisis del Trabajo Especial de Grado.
8. Migrar la información sobre los Trabajos Especiales de Grado que se encuentra almacenada en la base de datos de la Escuela de Artes a la ficha catalográfica del formato Marc21 de la BVMJM.

### **1.2.3. Alcance de la Aplicación**

El módulo de Trabajos Académicos es capaz de:

1. Realizar la búsqueda de Trabajos Académicos a través de los diferentes atributos que los caracterizan; por ejemplo: título, autor, tutor, fecha, palabras clave, etc.
2. Posibilitar la catalogación de los Trabajos Académicos usando el formato MARC 21.
3. Permitir la visualización y descarga de los Trabajos Académicos para usuarios registrados.
4. Permitir el acceso a los otros módulos de la Biblioteca, así como las otras secciones fuera de los módulos.
5. Permitir la administración de los Trabajos Académicos cargados en el portal mediante las funciones básicas de agregar, eliminar, actualizar y consultar.
6. Permitir la administración de usuarios, su registro y la asignación de perfiles y privilegios respecto al uso de las funcionalidades de los módulos.

### **1.3. Potenciales usuarios**

La BVMJM en general, así como el módulo de Trabajos Académicos en específico, se encuentra dirigida a musicólogos; sin embargo, se considera que será de gran utilidad para estudiantes y profesores de música, historiadores de música y arte en Venezuela, compositores, intérpretes, arreglistas y otras personas relacionadas con la investigación musical, así como también estudiantes, profesores y público en general que tengan interés sobre el arte venezolano.

## **CAPÍTULO II. Marco Teórico**

### **2.1. Bibliotecas Virtuales**

Iniciaremos por definir lo que sería el concepto de Biblioteca. El diccionario de la Real Academia de la Lengua Española define biblioteca como: “Institución cuya finalidad consiste en la adquisición, conservación, estudio y exposición de libros y documentos.”

Especificando el término de Biblioteca Virtual tenemos “*Se denomina biblioteca virtual a un ambiente digital cuyos materiales están en algún formato digital situados en Internet, de tal manera que pueden ser consultado desde cualquier parte del mundo, usando una computadora con conexión a la red*” (Teijero, 2006).

Haciendo una síntesis de las ventajas de una Biblioteca Virtual, según (Azocar, 2012, p. 104), podemos destacar:

- Constituir un aporte para entidades educativas y estudiantes que investigan en Internet, como fuente para sus trabajos escolares.
- Formar un depósito de obras que constituyen el acervo cultural de la humanidad, y en especial del mundo latino.
- Permitir engrandecer su contenido con aportes de escritores que quieran colocar en esta biblioteca virtual sus escritos.
- Dar oportunidad al visitante de conocer y valorar escritores y obras de distintos movimientos literarios, épocas y entornos socio-político-geográficos.
- Utilizar la Internet realmente como una herramienta, un importante recurso educativo en que se encuentre material bibliográfico para aplicar a la educación y formación del ser humano.

Crear en este sitio un lugar en donde los alumnos y usuarios puedan acceder a la información, resolver talleres, dejar las respuestas para su corrección, interactuar con sus profesores de forma síncrona y asíncrona.

### **2.2. Marco Conceptual**

En esta sección se tratarán de definir los conceptos que intervienen en el proceso de desarrollo del trabajo de investigación.

### **2.2.1. Módulo**

El diccionario de la Real Academia de la Lengua Española define módulo como: “Pieza o conjunto unitario de piezas que se repiten en una construcción de cualquier tipo, para hacerla más fácil, regular y económica”.

Para adaptar ese concepto a términos de un proyecto de programación podemos definirlo como: “En programación, un módulo es un software que agrupa un conjunto de subprogramas y estructuras de datos. Los módulos son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea, produciendo ahorro en los tiempos de desarrollo” (Alegsa, 2014). En el caso del portal, un módulo es un subprograma dentro del portal web llamado “Biblioteca Virtual Musicológica Juan Meserón” que funcionará independientemente, pero formará parte de un conjunto.

### **2.2.2. Trabajo Académico**

“El trabajo académico es un texto elaborado en torno a un tema y debe redactarse aplicando unas normas relativamente establecidas, en donde el destinatario de todo trabajo académico es siempre un lector culto, supuestamente ignorante del tema que vas a desarrollar”. (Universidad de Córdoba, s.f., p. 1)

### **2.2.3. Tesis**

Podemos definir tesis como "... un trabajo escrito de investigación, que se caracteriza por aportar conocimiento e información novedosa sobre un tema en particular en una extensión de más de 100 cuartillas. El objetivo es la comprobación de un planteamiento o la solución a un problema, a través de la investigación (bibliográfica o de campo) metódica y exhaustiva, a la que se le añade la reflexión sobre el tópico específico de la disciplina". (Universidad de las Américas Puebla, s.f. p. 1)

### **Trabajo de Asenso**

“El Trabajo de Ascenso es un aporte científico, técnico o cultural, que debe reunir los requisitos para ser considerado valioso en el campo de la especialidad o en áreas conexas y ser cónsono con la categoría a la cual aspira ascender el Profesor. Los Profesores Ordinarios aspirantes a ascender de categoría, elaborarán y presentarán el mencionado trabajo, de acuerdo a las condiciones establecidas en las presentes Normas”. (Universidad Nacional Experimental del Táchira, s.f., p. 1).

## 2.2.4. MARC 21

MARC es el acrónimo de *Machine Readable Catalogin* o “Catalogación legible por máquina”. Es una norma para el intercambio de información que permite estructurar e identificar los datos de tal forma que puedan ser reconocidos y manipulados por computadora. Este formato fue creado por el equipo de bibliotecarios de la Biblioteca del Congreso de Estados Unidos (Biblioteca del Congreso, 2014).

Ha sido diseñado para proveer especificaciones sobre la estructura con la que los datos serán comunicados entre sistemas de información cooperantes, no imponen pautas de almacenamiento interno, de visualización, identificación y ni descripción de los datos bibliográficos.

### Formatos de MARC 21

Los formatos o familias de formatos MARC 21 son de cinco clases diferentes, cada uno desarrollados para un uso en específico:

- Formato para Datos Bibliográficos: codifica los datos para describir, recuperar y controlar los diferentes tipos de materiales bibliográficos; por ejemplo: libros, publicaciones seriadas, recursos electrónicos, mapas, música, materiales visuales y materiales mixtos.
- Formato para Datos de Autoridad: codifica la información contenida en registros de autoridad de nombres, materias y series.
- Formato para Datos de Existencias: codifica información específica, como la signatura topográfica, cantidad de ejemplares y/o volúmenes, instituciones que posee un ítem, etc.
- Formato para Datos de Clasificación: codifica información contenida en un sistema de clasificación, por ejemplo: CDD, CDU o cualquier otro sistema que se desee automatizar.
- Formato para Datos de la Comunidad: codifica la descripción de recursos no bibliográficos que responden a las necesidades de información de una comunidad, como individuos, organizaciones, programas o servicios, eventos y otros recursos que pueden ayudar a los usuarios a conseguir la información que necesitan.

(*Classification Data*, 2000)

### ¿Por qué usar MARC 21?

“Se usa de manera estándar el formato MARC 21 debido a que sigue las normas del formato MARC, el cual evita la redundancia de los recursos de, en nuestro caso, la biblioteca. La decisión de utilizar MARC hace posible que las bibliotecas obtengan información catalográfica previsible y confiable. Si una biblioteca desarrollara un sistema propio que no utilizara registros MARC, no podría obtener las ventajas que ofrece una norma de amplia aplicación cuyo principal propósito es promover la transmisión e intercambio de la información”. (Biblioteca del Congreso, 2014).

### **2.2.5. Administrador de Contenido**

“Un Administrador de Contenidos (*Content Management System*, en inglés, abreviado CMS) permite la creación y administración de contenidos principalmente en páginas web.”

Consiste en una interfaz que controla una o varias bases de datos donde se aloja el contenido del sitio. El sistema permite manejar de manera independiente tanto el contenido como el diseño. Así, es posible manejar el contenido y darle en cualquier momento un diseño distinto al sitio sin tener que darle formato al contenido de nuevo, además de permitir la fácil y controlada publicación en el sitio a varios editores. Un ejemplo clásico es el de editores que cargan el contenido al sistema y otro de nivel superior que permite que estos contenidos sean visibles a todo público” (Investic, s.f.).

El Módulo Trabajos Académicos tendrá una serie de funcionalidades que serán para la generación de contenido dentro del módulo. Para eso se pensó en desarrollar un Administrador de Contenido que permitirá a un usuario con privilegios de administrador la actualización de contenidos en el portal web. En el caso del módulo Trabajos Académicos, los usuarios que posean el rango de administrador serán los encargados de, a través de una interfaz fácil de manejar, cargar las tesis y borrar las tesis subidas que, por alguna razón, no puedan o deban ser consultadas.

### **2.2.6. Buscador**

“Un buscador es una página de Internet que permite realizar búsquedas en la red. Su forma de utilización es muy sencilla, basta con introducir una o más palabras clave en una casilla y el buscador generará una lista de páginas web que se supone guardan relación con el tema solicitado. Digo se supone porque como veremos más adelante, esto no es siempre así”. (Universidad de Alicante, s.f.).

En el ámbito web los buscadores también se llaman motores de búsqueda.

#### **2.2.6.1. Metabuscador**

“Es un tipo de buscador que permite realizar varias búsquedas en diferentes motores de búsqueda respetando el formato de cada motor de búsqueda. Los resultados obtenidos son presentados bajo el formato del metabuscador utilizado”. (Universidad de Alicante, s.f.)

### **2.2.7. Base de Datos**

“Una base de datos o banco de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso. En este sentido; una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.” (Silberschatz, Fundamentos de Bases de Datos, 2014).

### **2.2.8. Sistema Manejador de Base de Datos**

Un Sistema Manejador de Bases de Datos (SMBD) consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos. La colección de datos, normalmente denominada base de datos, contiene información relevante para una empresa. El objetivo principal de un SMBD es proporcionar una forma de almacenar y recuperar la información de una base de datos de manera que sea tanto práctica como eficiente. (Silberschatz, 2011, p. 1).

Dentro de los sistemas manejadores de bases de datos de software libre más comunes y usados se encuentran PostgreSQL y MySQL.

## **CAPÍTULO III. Marco Metodológico y Herramientas de Desarrollo**

En este capítulo se describen los aspectos referentes al alcance de este proyecto de desarrollo del Módulo “Trabajos Académicos” y su integración al Portal BVMJM. Para eso se tomó la ejecución del trabajo de investigación previamente realizado, además de las metodologías implementadas en el Trabajo Especial de Grado.

### **3.1. Tipo de Investigación**

Existen varios tipos de investigación científica dependiendo del método y los fines que se persiguen. La investigación se define como “esfuerzo que se emprende para resolver un problema, claro está, un problema de conocimiento” (Sabino, 1992, p. 37). Encontramos que para este tipo de problema se realizó una investigación descriptiva, en la que se caracteriza un fenómeno o situación concreta indicando sus rasgos y características.

En el caso del módulo “Trabajos Académicos” de la BVMJM nos hicimos preguntas relacionadas con el desarrollo del tema, tecnologías a usar, metodologías de desarrollo, arquitectura sobre la cual va a correr el portal, características importantes que debe tener el módulo, entre otras.

### **3.2. Método de Desarrollo Ágil de Software**

“Los métodos de desarrollo ágil consisten en elementos individuales llamados prácticas. Las prácticas incluyen el uso de control de versiones, estándares de codificación entregando versiones del producto semanales a los interesados. Las mayorías de estas prácticas han existido durante años. Los métodos ágiles combinan de forma única aquellas prácticas que apoyan la filosofía ágil, descartando las demás, mezclándolas con ideas nuevas”. ( Shore y Warden, 2008, p. 9)

Los métodos de desarrollo ágil surgen a raíz de las fuertes desventajas en la ingeniería del software como dificultad al aceptar los cambios debido a altos costos, demanda de tiempo, documentación exhaustiva, sobre todo la re-planificación de un proyecto. Los métodos ágiles permiten un desarrollo iterativo facilitando las integraciones de nuevos requerimientos como también añadir nuevas funcionalidades como se muestra en la siguiente tabla.



Metodologías Ágiles	Metodologías Pesadas
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Menos énfasis en la arquitectura del software, La arquitectura se va definiendo y mejorando a lo largo del proyecto	La arquitectura del software es esencial y se expresa mediante modelos, Se promueve que la arquitectura se defina tempranamente en el proyecto
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
Las reuniones con el cliente son cortas donde se tratan temas puntuales de la etapa que se esté desarrollando en ese momento	El cliente interactúa con el equipo de desarrollo mediante reuniones largas donde se trata de manera de manera general todo el proyecto
Grupos pequeños (<10 integrantes)	Grupos grandes(>10 integrantes) y posiblemente distribuidos
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios, Se espera que no ocurran cambios de gran impacto durante el proyecto
Los roles son genéricos y flexibles	Los roles son muy específicos

*Tabla 1 Cuadro comparativo entre metodologías ágiles y tradicionales*

*Fuente: (Tellez, 2012)*

El 17 de febrero de 2001 diecisiete críticos de los modelos de mejora del desarrollo de software basados en procesos, convocados por Kent Beck, se reunieron en Snowbird, Utah para tratar sobre técnicas y procesos para desarrollar software, de dicha reunión surgieron 4 valores y 12 principios referentes al desarrollo de aplicación ágiles.

### **Valores:**

“Individuos e interacciones sobre procesos y herramientas.

Software funcionando sobre documentación extensiva.

Colaboración con el cliente sobre negociación contractual.

Respuesta ante el cambio sobre seguir un plan”. (Agile Manifesto, 2001)

## Principios:

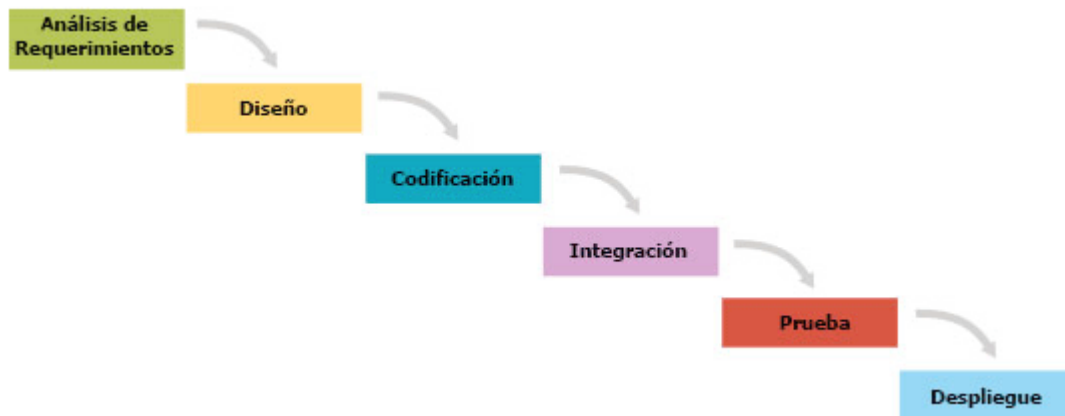
- “Nuestra mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
- Aceptamos que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
- Entregamos software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
- Los responsables de negocio y los desarrolladores trabajamos juntos de forma cotidiana durante todo el proyecto.
- Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
- El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
- El software funcionando es la medida principal de progreso.
- Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios debemos ser capaces de mantener un ritmo constante de forma indefinida.
- La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.
- La simplicidad, o el arte de maximizar la cantidad de trabajo no realizado, es esencial.
- Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.
- A intervalos regulares el equipo reflexiona sobre cómo ser más efectivo para a continuación ajustar y perfeccionar su comportamiento en consecuencia.”

(Agile Manifesto, 2001).

### 3.3. Metodología para el Desarrollo de la Aplicación

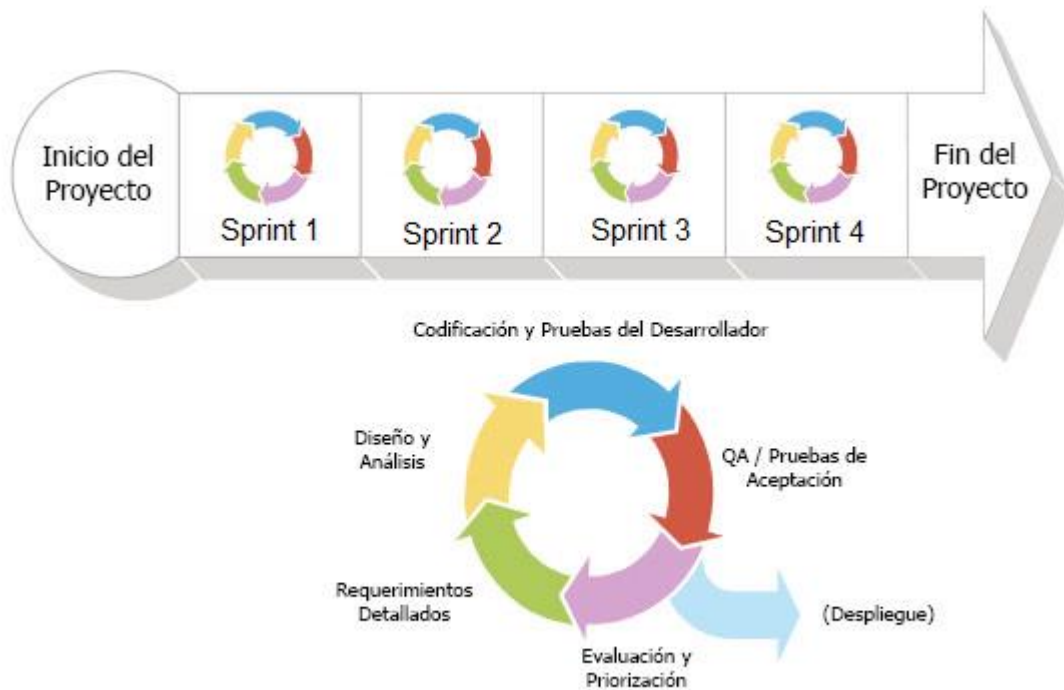
La metodología a utilizar en el desarrollo del Módulo de Carga, Consulta y Administración de Trabajos Académicos e Integración de Módulos del Portal, fue la metodología ágil Scrum, permitiéndonos crear un software de calidad, con la capacidad de responder a los cambios en los requisitos del cliente en cortas iteraciones, como también dar prioridad a requerimientos dependiendo de su importancia.

Scrum es un proceso ágil usado para gestionar y controlar desarrollos complejos de software y productos usando prácticas iterativas e incrementales, divide todo el proyecto en ciclos llamados *Sprints* que se realizan en periodos de tiempo de 2 a 3 semanas, donde cada *Sprint* comprende las actividades que se desarrollan en la Metodología Cascada generando un incremento (ver figura 1 y figura 2).



*Figura 1. Especificación de la Metodología Cascada.*

*Fuente: (James, Scrum Reference Card, 2014)*



*Figura 2. Proceso del desarrollo de un proyecto utilizando Scrum.*

*Fuente: (James, Scrum Reference Card, 2014)*

El mayor beneficio de Scrum se experimenta en el trabajo complejo que implica la creación de conocimiento y colaboración, tal como sucede en el desarrollo de nuevos productos. Scrum se asocia generalmente con el desarrollo de software orientado a objetos.

Según (James, Scrum Reference Card, 2014) los artefactos utilizados en Scrum son:

- **Pila de Productos (*Product Backlog*):** lista de requerimientos a implementar y que no se han asignado al Sprint a realizar, esos requerimientos se pueden priorizar de acuerdo a las necesidades del cliente.
- **Ítem de la Pila de Productos (*Product Backlog Item*):** Representa una función centrada en el cliente, que normalmente requiere de varias tareas para lograr la definición de hecho.
- **Lista de actividades de la corrida (*Sprint Backlog*):** lista de actividades que los desarrolladores deben realizar en el siguiente *Sprint*, es importante tener las actividades priorizadas con el fin de finalizar las de mayor valor al proyecto, las actividades de la corrida se

representa a menudo con un “radiador de información”, como una pizarra de tareas (ver Figura 12).

- **Incremento:** es la suma de cada requerimiento finalizado de la pila de productos, este requerimiento está en condiciones de uso.
- **Gráfica de trabajos faltantes (*Sprint Burndown Chart*):** gráfico en donde se aprecia las el trabajo restantes de un proyecto basado en el tiempo, en Scrum es actualizado después de la finalización de cada *Sprint*, dando una vista simple de los progresos de cada *Sprint* (ver Figura 13).

Para Scrum existen tres roles básicos:

- **Cliente (*Product Owner*):** son los interesados en el desarrollo del proyecto, es quien tiene la visión de lo que desea y debe transmitirla a los miembros del equipo, en este caso, (Profesora Mariantonia Palacios).
- **Facilitador (*Scrum Master*):** líder del equipo, encargado de ayudar a solventar los obstáculos con el fin de lograr el objetivo, representado por la tutora, (Profesora Yusneyi Carballo).
- **Equipo de Desarrollo (*Development Team*):** grupo de personas que de manera grupal desarrollan el proyecto (representado por Víctor Cardozo).

## Fases de las corridas en Scrum

Según (James, Scrum Reference Card, 2014) las fases de las corridas en Scrum son:

### Reunión de Planificación de Sprint (*Sprint Planning Meeting*):

Al inicio de cada *Sprint*, el cliente y el equipo tienen una Reunión de Planificación del *Sprint* donde negocian qué ítems de la pila de productos intentarán convertir en producto funcionando durante el *Sprint*. El cliente es el responsable de declarar cuáles son los ítems más importantes para el negocio. El equipo es responsable de seleccionar la cantidad de trabajo que cree que podrán realizar sin acumular deuda técnica. El equipo "toma" el trabajo desde la pila de productos hacia la lista de actividades de la corrida.

### Reunión Diaria del Scrum (*Daily Scrum*)

Es una reunión corta de aproximadamente 15 minutos que se realiza diariamente, donde participan todos los miembros de desarrollo, cada miembro del equipo resume lo que hizo el día

anterior, lo que hará hoy, y qué impedimentos está enfrentando. También asegurar de que todo está en camino y todos los miembros del equipo tienen las herramientas que necesitan. Es una herramienta esencial para tener un flujo de información libremente entre los miembros del equipo.

### **Ejecución del *Sprint***

Al equipo puede resultarle útil mantener una lista de Tareas del Sprint (*Sprint Task List*), un gráfico de trabajos faltantes, y un Listado de Impedimentos. Durante la ejecución del Sprint es común descubrir tareas adicionales necesarias para alcanzar las metas del *Sprint*. Los impedimentos causados por los problemas que escapan al control del equipo se consideran impedimentos organizacionales.

### **Reunión de Revisión del Sprint**

Después de la ejecución de Sprint, el equipo mantiene una reunión de revisión del Sprint para mostrar un incremento del producto al Cliente y a todos los demás interesados.

La reunión debe ser una demostración en vivo de los avances realizados durante el *Sprint*. Después de la demostración, el Cliente revisa los compromisos contraídos en la Reunión de Planificación del *Sprint* y declara qué ítems considera terminados.

### **Reunión de Retrospectiva**

Cada *Sprint* finaliza con una retrospectiva. En esta reunión el equipo reflexiona sobre su propio proceso. Inspeccionan su comportamiento y adoptan las medidas para adaptarlo a en los futuros *Sprints*. El propósito de la retrospectiva es realizar una mejor continua del proceso.

## **3.4. Tecnologías para el desarrollo de la aplicación.**

Para el desarrollo y correcto cumplimiento de los utilizamos las siguientes tecnologías para el desarrollo de la aplicación.

### **3.4.1. PHP (PHP Hypertext Pre-processor)**

Es un lenguaje de programación de código abierto usado para el desarrollo de tecnologías web en conjunto con HTML. Se utiliza principalmente para la generación de páginas web dinámicas, ejecución de scripts desde la línea de comando o la generación dinámica de archivos (PHP, 2014).

Entre las ventajas del uso de este lenguaje tenemos:

- Es rápido, estable, seguro, fácil de usar y de código abierto.
- El código PHP se inserta directamente en el código HTML de un sitio web.
- Es fácil de entender y aprender.
- Es un lenguaje multiplataformas.
- PHP no requiere mucho uso del sistema o recursos.
- Su estabilidad está sustentada gracias al aporte de la comunidad de programación que lo utiliza.
- Capacidad de conexión con diversas plataformas de Base de Datos.
- Algunas de sus desventajas:
- La legibilidad del código puede verse afectada al mezclar sentencias HTML y PHP.
- Se necesita la ayuda de *frameworks* para ayudar a controlar toda la seguridad si no se es experto.

### 3.4.2. CakePHP

Es un *framework* libre, de código abierto para el desarrollo de aplicaciones web escritas en PHP. Su objetivo principal es permitir trabajar de forma estructurada, rápida y sin pérdida de flexibilidad, el cual logra trabajando bajo el patrón de diseño Modelo Vista controlador (MVC) y uso de la técnica de programación Mapeo Objeto-Relacional. (CakePHP, 2014).

El ORM (*Object-Relational Mapping* o Mapeo Objeto-Relacional) se puede definir como “...es una técnica de programación para convertir datos entre el lenguaje de programación orientado a objetos utilizado y el sistema de base de datos relacional utilizado en el desarrollo de nuestra aplicación.” (Programación, 2013).

#### Ventajas del ORM.

Según (Programación, 2013) las ventajas del ORM son:

1. “Rapidez en el desarrollo. La mayoría de las herramientas actuales permiten la creación del modelo por medio del esquema de la base de datos, leyendo el esquema, nos crea el modelo adecuado.
2. Abstracción de la base de datos. Al utilizar un sistema ORM, lo que conseguimos es separarnos totalmente del Sistema de Base de datos que utilizemos, y así si en un futuro

debemos de cambiar de motor de bases de datos, tendremos la seguridad de que este cambio no nos afectará a nuestro sistema, siendo el cambio más sencillo.

3. Reutilización. Nos permite utilizar los métodos de un objeto de datos desde distintas zonas de la aplicación, incluso desde aplicaciones distintas.
4. Seguridad. Los ORM suelen implementar sistemas para evitar tipos de ataques como pueden ser los SQL *injections*.
5. Mantenimiento del código. Nos facilita el mantenimiento del código debido a la correcta ordenación de la capa de datos, haciendo que el mantenimiento del código sea mucho más sencillo.
6. Lenguaje propio para realizar las consultas. Estos sistemas de mapeo traen su propio lenguaje para hacer las consultas, lo que hace que los usuarios dejen de utilizar las sentencias SQL para que pasen a utilizar el lenguaje propio de cada herramienta”.

CakePHP es distribuido bajo la licencia Expat, algunas veces se le llama «MIT License» (GNU, 1999).

Entre las principales características de CakePHP se encuentran:

- Compatible con versiones de PHP 5.2.6 y superiores, lo que permite que permitan trabajar en prácticamente todos los servidores comerciales que manejen PHP y no tengan necesariamente la última versión.
- Incorporación del patrón de diseño Modelo Vista Controlador (MVC) para facilitar el desarrollo separando lo que es la vista del manejo de datos y la lógica de la aplicación.
- Mapeado de objetos a bases de datos relacionales (ORM) que nos da una capa más de seguridad en la base de datos a posibles ataques al servidor.
- Operaciones CRUD para interactuar con la base de datos (*Create, Read, Update, Delete*) que son funciones básicas esenciales para muchas aplicaciones que manejen bases de datos.
- Generación automática de código básico en los controladores y vistas en las aplicaciones mediante la consola de CakePHP que nos ahorra tiempo en el desarrollo de funciones básicas sobre las tablas de la base de datos.
- Componentes de seguridad, manejo de sesiones y de peticiones.
- Listas de control de acceso flexibles.
- Integración con otras herramientas a través de *plugins*.
- Funciones de validación de formularios.



## **Patrón Modelo Vista Controlador (MVC) en CakePHP**

“Modelo Vista Controlador es un patrón para diseño de software que ayuda a separar lógicamente el código haciéndolo reusable y mantenible. Fue ideado por Trygve Reenskaug entre 1978 y 1979 mientras trabajaba en la compañía Xerox y fu desarrollado para ser implementado en SmallTalk-80. Programar utilizando MVC consiste en separar la aplicación en tres partes principales. El modelo representa los datos de la aplicación, la vista hace una presentación del modelo de datos, el controlador maneja y enlaza las peticiones realizadas por los usuarios” (CakePHP, 2014).

En CakePHP, el patrón MVC, abarca:

### **Modelo:**

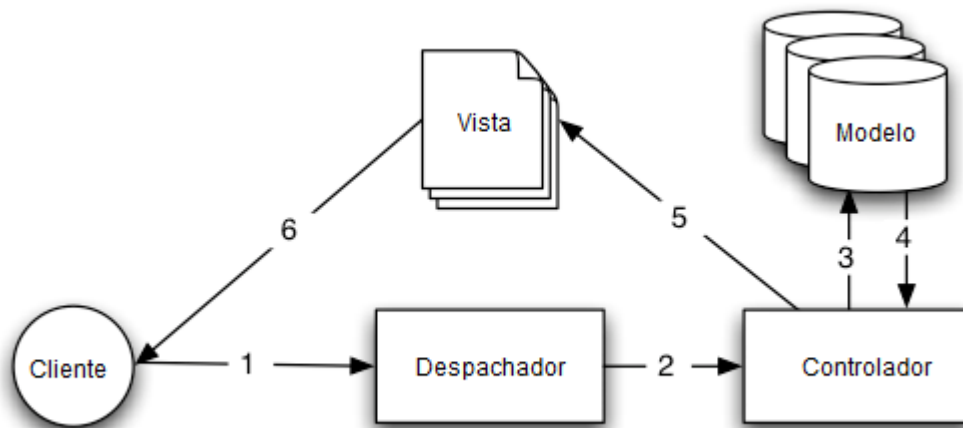
Representa la parte de la aplicación que implementa la lógica de negocio, esto significa que es responsable de la recuperación de datos, así como su procesamiento, validación, asociación y cualquier otra tarea relativa a la manipulación de dichos datos. Un modelo representa una tabla perteneciente de la base de datos.

### **Vista:**

La vista hace una presentación de los datos del modelo. Estando separada de los objetos del modelo, es responsable del uso de la información para producir cualquier interfaz de presentación de cualquier petición que se presente. No solo se limita a HTML o texto que represente los datos, también puede mostrar videos, música, documentos y cualquier otro formato a imaginar.

### **Controlador:**

La capa del controlador gestiona las peticiones de los usuarios, encargado de responder la información solicitada con la ayuda tanto del modelo como de la vista. Son vistos como administradores, controlan el flujo y lógica de la aplicación. Cada petición web es dirigida por un controlador, es cual contiene llamadas a métodos para después pasar la respuesta a la vista (ver Figura 3).



*Figura 3. Típica petición en CakePHP usando MVC*

*Fuente: (CakePHP, 2014)*

El ciclo de una petición típica en CakePHP comienza cuando un usuario solicita una página o un recurso de tu aplicación. Esta solicitud es procesada por un despachador que selecciona el controlador correcto para manejarlo.

Una vez que la solicitud llega al controlador, este se comunicará con la capa del Modelo para cualquier proceso de captación de datos o el guardado de los mismos según se requiera. Una vez que finaliza esta comunicación, el controlador procederá a delegar en el objeto de vista correcto la tarea de generar una presentación resultante de los datos proporcionada por el modelo.

Finalmente, cuando esta presentación se genera, se envía de inmediato al usuario. Casi todas las solicitudes para la aplicación van a seguir este patrón básico.

### **¿Por qué utilizar MVC?**

Debido a que es un patrón de diseño de software ampliamente probado que convierte una aplicación en un paquete modular, fácil de mantener, y mejora la rapidez del desarrollo. La separación de las tareas de tu aplicación en modelos, vistas y controladores hace que la aplicación que se está desarrollando sea mantenible.

El diseño modular también permite a los desarrolladores y los diseñadores trabajar simultáneamente, incluyendo la capacidad de hacer prototipos rápidos, como también realizar cambios en la aplicación si afectar a los demás.

### 3.4.3. MySQL

MySQL, es el Sistema de Gestión de Base de datos Relacional, multihilo, multiusuario y Open Source más popular de SQL, ha sido desarrollado, distribuido y soportado por Oracle Corporation. (MySQL, 2014)

MySQL nos permite añadir, acceder y procesar los datos almacenados en una base de datos informática, necesita un Sistema de Gestión de Base de Datos como MySQL Server. Dado que los ordenadores son muy buenos en el manejo de grandes cantidades de datos, sistemas de gestión de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones. (dev.mysql.com, 2013)

MySQL es la base de datos de código abierto más popular del mundo con más de 15 millones de instalaciones estimadas y decenas de miles de nuevas descargas cada día. Se trata de la tercera base de datos más utilizada en el mundo, después de Oracle y Microsoft SQL Server. Cuenta con el rendimiento y la escalabilidad para satisfacer las necesidades de la mayoría de las aplicaciones.

### 3.4.4. Otras herramientas para el desarrollo

Existen otros tipos de herramientas muy utilizadas para el desarrollo de sistema web pero no obligatorias. Una de la más importantes y de gran ayuda son los IDE's (*Integrated Development Environment* / Entorno de Desarrollo Integrado). Los IDE's son, según (PC Mag, s.f.) "Entornos de desarrollo que superan en capacidades y prestaciones a los editores de texto para escribir los códigos de programación. Entre sus principales componentes tiene: Editor de texto, depurador de código, acceso al control de versiones, autocompletado de código, el intérprete, compilador entre otros componentes que varían dependiendo del tecnología de desarrollo". Entre los más comunes están:

- Eclipse
- NetBeans
- Visual Studio

Aunque los IDE's suelen ser de gran ayuda para el desarrollo, estos generalmente requieren muchos recursos del computador, pudiendo reducir el rendimiento del mismo, los autores prefieren

un editor rápido y sencillo, en lugar de utilizar un IDE. Para el desarrollo del Trabajo Especial de Grado utilizará un editor de texto simple: Sublime Text 3 (ver Anexo A).

Otra herramienta muy utilizada es el Sistema de Control de Versiones o SCM (siglas del inglés *Source Code Management*). "El control de versiones es un sistema que registra los cambios realizados sobre un archivo o conjunto de archivos a lo largo del tiempo, de modo que puedas recuperar versiones específicas más adelante" (Git, 2009)

Entre los más comunes, están:

- SVN (*Subversion*)
- GIT
- Mercurial.

Para controlar el proceso de desarrollo Scrum existen diversas herramientas que nos dan el estado actual de un sprint e información detallada de los artefactos en un momento determinado. Este tipo de herramientas otorgan un mejor control de la metodología sin la documentación extensa y facilita el uso y respeto de las normas de Scrum. Entre las principales herramientas para el manejo del proceso Scrum gratuitas tenemos:

- PangoScrum
- Kunagi
- IceScrum
- ScrumDo

Adicionalmente para la administración de la base de datos se utilizara la herramienta MySQL Workbench.

Debido a la gran cantidad de herramientas y plataformas es necesario estudiarlas y analizarlas, para seleccionar aquella que mejor se adapte al desarrollo del Portal Biblioteca Virtual Musicológica Juan Meserón.

Para garantizar una uniformidad en el desarrollo de este módulo con respecto a los otros módulos del portal se utilizarán.

- CakePHP como *framework*.
- MySQL como Sistema de Bases de Datos.
- Sublime Text 3 como editor de texto por ser rápido y sencillo.

- GIT para el control de versiones debido a su facilidad de uso y su eficacia.
- Kunagi para el control del desarrollo en Scrum.

### 3.5 Implementación del Formato MARC 21

Entre los requerimientos más importantes para este proyecto de Biblioteca Virtual están la utilización del formato MARC 21 para la catalogación de los Trabajos Académicos. De acuerdo a (Mendoza, 2014) cuando se catalogan trabajos académicos se utilizan los siguientes campos.

#### 3.5.1. Cabecera

En la cabecera se utilizan los siguientes campos.

Desc.	Long del reg					Er	Tr	Nb	Tc	Ec	Li	Lc	Posic inic de los camp				
Posic.	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16
Cont.						N	t	M	#	#	2	2					

Desc.	Nb	Nc	Rm	Lc	Pi	Ic	Nd
Posic.	17	18	19	20	21	22	23
Cont.	7	a	#	4	5	0	0

#### 05 - Estado del registro

- a - Aumentado el nivel de codificación
- c - Corregido o revisado
- d - Suprimido
- n - Nuevo
- p - Aumentado el nivel de codificación utilizado antes de la publicación

#### 06 - Tipo de registro

- a - Material textual
- c - Música notada
- d - Música notada manuscrita
- e - Material cartográfico
- f - Material cartográfico manuscrito

- g - Material gráfico proyectable
- i - Grabación sonora no musical
- j - Grabación sonora musical
- k - Material gráfico bidimensional, no proyectable
- m - Archivo de ordenador
- o - Kit
- p - Material mixto
- r - Objeto tridimensional artificial o natural
- t - Material textual manuscrito

## **07 - Nivel bibliográfico**

- a - Parte componente monográfica
- b - Parte componente seriada
- c - Colección
- d - Subunidad
- i - Recurso integrable
- m - Monografía
- s - Publicación seriada

## **17 - Nivel de codificación**

- # - Nivel completo
- 1 - Nivel completo, material no examinado
- 2 - Nivel incompleto, material no examinado
- 3 - Nivel abreviado
- 4 - Nivel básico
- 5 - Nivel parcial (preliminar)
- 7 - Nivel mínimo
- 8 - Nivel de prepublicación
- u - Desconocido
- z - No aplicable

## **18 - Forma de catalogación descriptiva**

- # - No es ISBD
- a - AACR 2

c - ISBD sin puntuación

i - ISBD con puntuación

u – Desconocida

### 3.5.2. Campos de Longitud Fija

Para Trabajos Académicos se utilizan los siguientes campos de longitud fija.

Desc.	Fecha creación						Tf	Fecha 1				Fecha 2				País		
Posic.	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17
Cont.							s	#	#	#	#	#	#	#	#	x	x	#

Desc	Ilustr.				Aud	Fi	Nat contenido				Pg	Pc	H	I	FI	B
Posic.	18	19	20	21	22	23	24	25	26	27	28	29	30	31	33	34
Cont.	#	#	#	#	#	#	b	m	#	#	#	0	0	0	0	#

Desc	Lengua			Rm	Fc
Posic.	35	36	37	38	39
Cont.	u	n	d	#	#

### 06 - Tipo de fecha/estado de la publicación

b - No consta información; implica fechas a.C.

c - Recurso continuado con publicación en curso

d - Publicación cerrada

e - Fecha detallada

i - Fechas comprendidas en una colección

k - Rango de años del grueso de la colección

m - Fechas múltiples

n - Fecha desconocida

p - Fechas de distribución/estreno/edición y de sesión de producción/ grabación cuando difiere

q - Fecha dudosa

r - Fechas de la reimpresión/reedición y del original

- s - Fecha única conocida/probable
- t - Fechas de publicación y de copyright
- u - Estado desconocido
- | - No se utiliza

## **07-10 - Primera fecha**

- 1-9 - Dígitos de la fecha
- # - El elemento fecha no es aplicable
- u - Fecha total o parcialmente desconocida
- ||| - No se utiliza

## **11-14 - Segunda fecha**

- 1-9 - Dígitos de la fecha
- # - El elemento fecha no es aplicable
- u - Fecha total o parcialmente desconocida
- ||| - No se utiliza

## **15-17 - Lugar de publicación, producción o ejecución**

Lista de códigos MARC para países

## **24-27 - Naturaleza del contenido**

- # - No se especifica la naturaleza del contenido
- a - Resúmenes/sumarios
- b - Bibliografías
- c - Catálogos
- d - Diccionarios
- e - Enciclopedias
- f - Manuales
- g - Artículos sobre temas legales
- i - Índices
- j - Documentos de patente
- k - Discografías



l - Legislación  
m - Tesis  
n - Estado de la cuestión en una materia  
o – Reseñas

## **35-37 - Lengua**

Lista de códigos MARC para idiomas

### **3.5.3. Campos de Longitud Variable**

- 040 ## Fuente de la catalogación (NR)
- 041 0# Código de lengua (R)
- 044 ## Código del país de la entidad editora/productora (NR)
- 082 0# Número de la Clasificación Decimal Dewey (R)
- 092 ## Clasificación local
- 100 1# Punto de acceso principal - Nombre de persona (NR)
- 260 ## Publicación, distribución, etc. (pie de imprenta) (R)
- 245 10 Mención de título (NR)
- 300 ## Descripción física (R)
- 500 ## Nota general (R)
- 502 ## Nota de tesis (R)
- 504 ## Nota de bibliografía, etc. (R)
- 505 00 Nota de contenido con formato (R)
- 546 ## Nota de lengua (R)
- 588 ## Nota de fuente de la descripción (R)
- 600 10 Punto de acceso adicional de materia - Nombre de persona (R)
- 610 20 Punto de acceso adicional de materia - Nombre de entidad (R)
- 650 #0 Punto de acceso adicional de materia – Término de materia (R)
- 651 #0 Punto de acceso adicional de materia - Nombre geográfico (R)
- 653 #: Término de indización – No controlado (R)
- 700 1# Punto de acceso adicional - Nombre personal (R)
- 710 2# Punto de acceso adicional - Nombre corporativo (R)
- 740 :# Punto de acceso adicional - Título relacionado o analítico no controlado (R)

- 850 ## Institución que posee los fondos (R)
- 852 :: Localización (R)
- 856 :: Localización y acceso electrónicos (R)

### 3.5.4 Ejemplo

En el siguiente ejemplo se visualiza algunos de los campos del formato MARC 21, más específicamente los campos obligatorios y algunos opcionales, asociados a un Trabajo Académico

#### Formato MARC21

```
=h-005 n
=h-006 t
=h-007 m
=h-017 7
=h-018 a
=008 150220cyymddyymddxxx#und
=040 ##$aCentro de Catalogación de la Escuela de Artes
=044 ##$ave
=082 0#$a111$b1
=092 ##$a122$b122$c122
=100 1#$aVictor Cardozo
=245 10$aAnálisis de requerimientos modulo de trabajos académicos BVMJM$bSeminario Módulo Trabajos Académicos BVMJM$cLicenciado en Computación$hSeminario
=260 ##$aCaracas$bYusneyi Carvallo$c10-11-2014
=300 ##$a100$bninguna$c10x12x12$eno
=500 ##$a18
=546 ##$a18
=650 #0$aComputación
=653 ##$atag1, tag2, tag3
=041 ##$aes$b$es$hes
=502 ##$a18
=504 ##$a18
```

*Figura 4. Formato MARC 21*

*Fuente: (Autor, 2014)*

## CAPÍTULO IV. Marco Aplicativo

Este capítulo es de suma importancia ya que se presentan las funcionalidades del módulo de Trabajos Académicos, así como también perfiles de usuarios, los *sprints* realizados cumpliendo con la metodología Ágil Scrum y las pruebas realizadas a la aplicación. El módulo de Trabajos Académicos de la Biblioteca Virtual Musicológica Juan Meserón proporciona acceso libre y gratuito a este tipo de documentos dependiendo del tipo de derecho de autor bajo el cual esté regida el documento, definido previamente por su respectivo autor o autores.

### 4.1. Perfiles de usuarios

A continuación se presentan los distintos tipos de usuarios que pueden acceder al portal

- **Visitante no registrado:** Este tipo de usuario está caracterizado por no poseer una cuenta registrada en el portal. Este usuario puede visualizar las páginas informativas del sitio web, leer, los documentos sin necesidad de tener una cuenta activa en el portal.
- **Visitante registrado:** Este tipo de usuario está caracterizado por tener una cuenta de usuario registrada, además de poder hacer todas las actividades que pueden realizar el visitante no registrado, tiene la opción de crear su biblioteca personalizada y tanto descargar como imprimir los trabajos cargados en el módulo.
- **Catalogador:** Entre los visitantes registrados podemos tener visitantes con el rol de catalogador. Los usuarios con este rol además de poder hacer todas las actividades que pueden realizar los visitantes registrados este usuario puede cargar, modificar y eliminar cualquier tipo de obra catalogada con el formato MARC 21.
- **Administrador:** Además del rol de catalogador también existe el rol de administrador. Los usuarios con este rol además de poder realizar todas las actividades del catalogador también pueden crear, consultar, modificar usuarios, enlaces, campos de la ficha catalográfica, preguntas frecuentes, entre otros; modificar contenidos que se reflejará en la biblioteca, realizar respaldo de la base de datos y realizar respaldo de la aplicación.

## **4.2. Requerimientos del módulo desarrollado y de la BVMJM**

En base a los requerimientos solicitados por el cliente, representado por la profesora Mariantonia Palacios, se desarrollaron las siguientes funcionalidades.

1. Listar los trabajos asociados al módulo de Trabajos Académicos, que coincidan con los criterios de búsqueda especificados en el filtro.
2. Visualizar, descargar e imprimir los Trabajos Académicos cargados en el portal.
3. Visualizar una ficha informativa de los Trabajos Académicos consultados mediante los filtros o la búsqueda.
4. Clasificar los Trabajos Académicos por materia, título, autor, fecha y tutor.
5. Funcionalidad de buscador simple y avanzado donde el usuario puede localizar de manera sencilla en función de palabras clave en los campos de título, autor, tutor, fecha o materia.
6. Permitir al administrador y al catalogador agregar Trabajos Académicos utilizando el estándar MARC 21.
7. Permitir al administrador y al catalogador modificar la información de los trabajos cargados en el sistema.
8. Permitir al administrador y al catalogador eliminar los trabajos cargados en el sistema.
9. Permitir al usuario registrado descargar, imprimir, agregar y eliminar Trabajos Académicos de su biblioteca personal.
10. Permitir al usuario el acceso a los otros módulos de la aplicación a través de enlaces.

## **4.3. Requisitos del Sistema**

A continuación se muestran las características en el punto de vista de requisitos funcionales y no funcionales del portal.

### **4.3.1. Requisitos Funcionales**

Los requisitos funcionales definen una función del sistema de software o sus componentes. Una función es descrita como un conjunto de entradas, comportamientos y salidas. Los requerimientos funcionales pueden ser: cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone, el sistema debe cumplir. Los requerimientos de comportamiento para cada requerimiento funcional se muestran en los casos de uso.

Entre los requisitos funcionales identificados para el desarrollo del portal tenemos

- Permitir al usuario catalogador y administrador agregar Trabajos Académicos al módulo usando el sistema de catalogación MARC 21 o eliminarlos según sea necesario.
- Capacidad de modificar detalles en la catalogación de los trabajos expuestos en el portal.
- Diferenciación entre los diferentes roles (administrador, usuario registrado, usuario sin registrar, catalogador)
- Capacidad de imprimir y descargar los trabajos expuestos de los autores que previamente hayan concedido los derechos de distribución de su trabajo.
- Buscar información interna dentro del portal mediante palabras clave.
- Panel de administración que permita agregar, eliminar y modificar los roles de los usuarios registrados.

#### 4.3.2. Requisitos no Funcionales

Los requisitos no funcionales describen aspectos del sistema de software o sus componentes que no implican una relación directa con el comportamiento funcional del sistema.

Entre los requerimientos no funcionales que buscamos implementar están:

- Hacer eficientes las consultas a la base de datos, tanto en la data extraída como en cantidad de consultas necesarias.
- Optimizar la carga de documentos a la hora de la visualización en línea.
- Diseñar una interfaz tomando en cuenta los principios de usabilidad y accesibilidad.
- Utilizar el *caching* de CakePHP para reducir los tiempos de espera.
- Emplear validaciones del lado del cliente primero para evitar llamadas innecesarias al servidor donde igual se haría una segunda validación.

#### 4.4. Definición y análisis del modelo de datos

Para el diseño de la base de datos se realizó especial énfasis en el cumplimiento de los convenios del framework CakePHP, estos convenios definen “reglas” que permiten aprovechar sus funcionalidades al máximo. Entre estos convenios destacan:

- El nombre de las tablas debe estar en plural y en idioma inglés. La razón es que CakePHP no reconoce todas las variantes y reglas de plurales del castellano, por el contrario reconoce todas las reglas de plurales del idioma inglés, de tal modo que el modelo “*User*” o “*UserName*” (si son dos palabras) se corresponde con la tabla “*users*” o “*user\_name*” respectivamente.

- Las claves foráneas o ajenas (*foreign keys*) en las relaciones ‘a muchos’ (*hasMany*), ‘pertenece a’ (*belongsTo*) y ‘a uno’ (*hasOne*) se reconocen por defecto si el nombre del campo se escribe usando el singular de la tabla con la que se relaciona y terminando en *\_id*. Por ejemplo el modelo “*Department*” tiene una relación ‘a muchos’ con el modelo “*Employee*”. En la tabla “*employees*” escribiremos un campo con el nombre “*employee\_id*”. En caso de que el nombre de la tabla tenga varias palabras, como en “*category\_types*”, la clave sería “*category\_type\_id*”.
- Todas las tablas deben tener un campo “*id*”, así se trate de una tabla relación

## Tablas de catalogación

**Nombre de tabla:** trabajo (*items*).

**Descripción:** tabla para almacenar los campos usados en la catalogación de manera estructural. Aquí se almacenan los campos de longitud fija y campos de longitud variable para cada tipo de obra que presenta la Biblioteca Virtual “Juan Meserón”.

## Tablas de Contenidos del Sitio

**Nombre de tabla:** páginas estáticas (*pagetexts*).

**Descripción:** tabla en la que se almacena información relaciona con las páginas estáticas o de contenido de la Biblioteca.

**Nombre de tabla:** mensajes y comentarios (*messages*).

**Descripción:** tabla en la que se guarda los comentarios y mensajes de usuarios hacia el personal de la Biblioteca.

**Nombre de tabla:** búsquedas (*searches*).

**Descripción:** tabla en la que se almacena la dirección ip, el usuario, la fecha y la búsqueda como tal que hizo el usuario.

**Nombre de tabla:** Noticias (*news*).

**Descripción:** tabla en la que se almacenan los diferentes perfiles de usuario.

**Nombre de tabla:** enlaces (*links*).

**Descripción:** tabla en la que se guarda los enlaces del sitio.

**Nombre de tabla:** preguntas frecuentes (*faqs*).

**Descripción:** tabla en la que se guarda los enlaces del sitio.

**Nombre de tabla:** eventos (*events*).

**Descripción:** tabla en la que se guarda los diferentes eventos que se puedan realizar.

### **Tablas de Seguridad y Autenticación**

**Nombre de tabla:** inicio de sesión (*logins*).

**Descripción:** tabla en la que se almacena información como dirección ip, usuario, fecha, acción realizada, y el controlador donde el usuario accedió. Tabla utilizada para almacenar información que estará asociada directamente con el *log* de visitas.

**Nombre de tabla:** perfiles de usuario (*profiles*).

**Descripción:** tabla en la que se almacenan los diferentes perfiles de usuario.

**Nombre de tabla:** grupo o rol (*groups*).

**Descripción:** tabla en la que se guarda el rol del usuario

## **4.5. Principales interfaces de la Biblioteca Virtual Musicológica “Juan Meserón”**

A continuación se presenta las propuestas de interfaz para el TEG.

### **4.5.1. Página principal de la Biblioteca Virtual**

La figura 5 muestra la página principal de la Biblioteca Virtual Musicológica “Juan Meserón”.

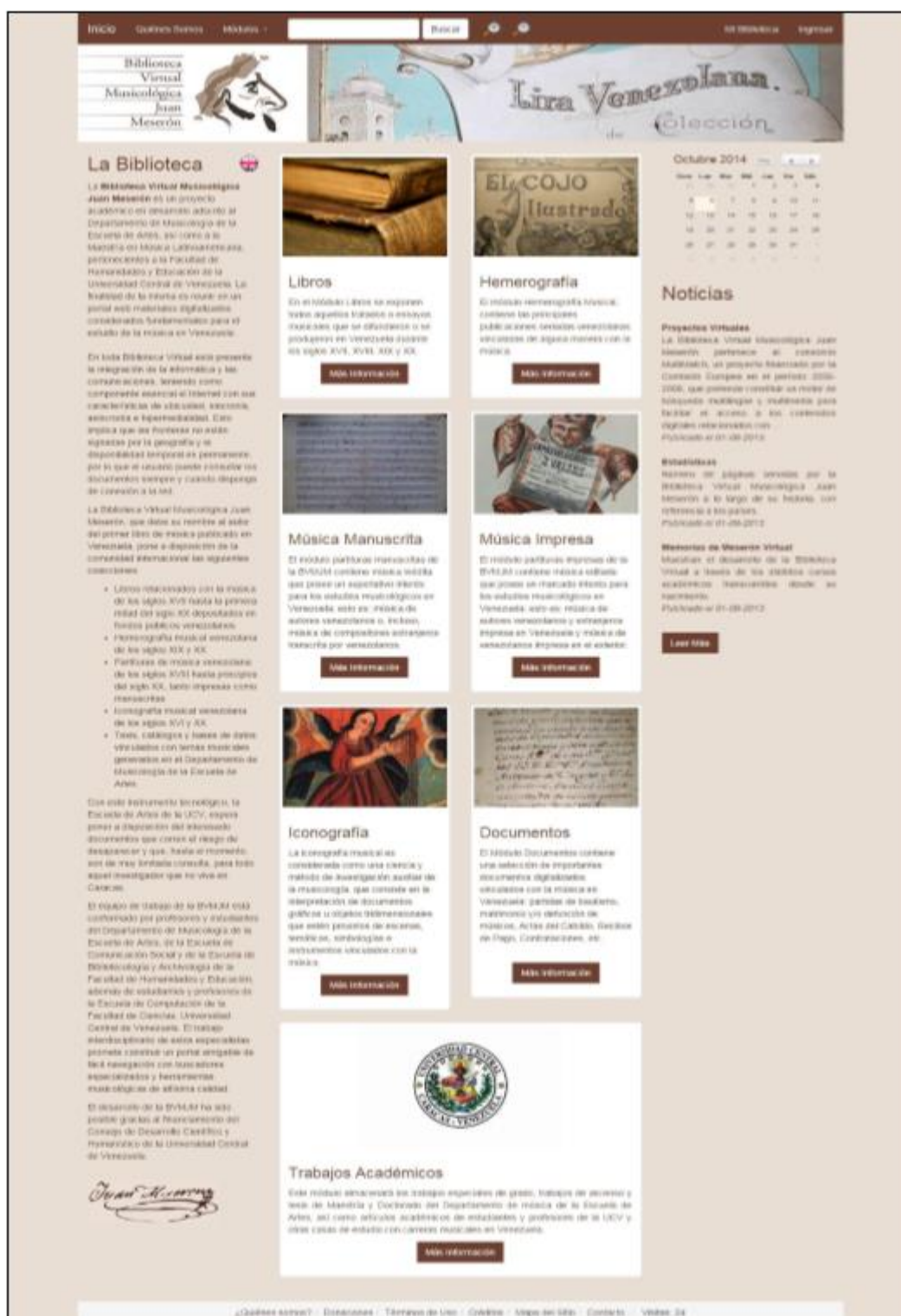


Figura 5. Página principal de la Biblioteca Virtual Musicológica Juan Meserón

Fuente: (Autor 2015)



- **Inicio:** permite a todo tipo de usuario acceder a la página principal de la Biblioteca Virtual Musicológica “Juan Meserón”.
- **Quiénes somos:** al hacer clic allí, se muestra una página informativa sobre el proyecto de la Biblioteca Virtual Musicológica Juan Meserón.
- **Módulos:** se muestra un menú vertical con los diferentes módulos que constituyen la Biblioteca Virtual Musicológica Juan Meserón.
- **Un buscador simple:** se podrá encontrar cualquier obra, según lo que se escriba en el buscador.
- **Mi Biblioteca:** donde el sólo el usuario registrado podrá hacer uso de esta sección.
- **Ingresar:** permite al usuario ingresar con el perfil que tenga, o registrarse si aún no lo ha hecho.
- **Noticias:** muestra información de interés.
- **Calendario:** muestran la fecha actual de la visita a la biblioteca, así como también mostrar eventos si existen.
- **Pie de página:** muestra enlaces a información importante y funcionalidades complementarias como donaciones, términos de uso, créditos, mapa del sitio, contacto y contador de visitas.
- **Logos:** se encuentran los logos de las instituciones involucradas en el desarrollo de la Biblioteca. Al seleccionar algunos de los logos podrán acceder a la página principal de esa institución.

Se observa también, una breve descripción de cada módulo, así como también los accesos a ellos.

#### 4.5.2. Interfaces de administrador.

Página de ingreso al módulo administrador Toda persona que haga uso del perfil de administrador debe realizar el ingreso al módulo administrador, accediendo a la vista que se observa a continuación:

The image shows a web browser window displaying the login page of a virtual library. The header is dark brown with white text for navigation: 'Inicio', 'Quiénes Somos', 'Módulos', a search bar with a 'Buscar' button, 'Mi Biblioteca', and 'Ingresar'. Below the header is a banner with a stylized illustration of a person and the text 'Biblioteca Virtual Musicológica Juan Meserón' and 'Lira Venezolana Colección'. The main content area is light beige and titled 'Ingreso de Usuario'. It contains two input fields: 'Nombre de Usuario' and 'Contraseña'. Below these are three buttons: 'Ingresar' (dark brown), 'Registrarse' (light brown), and 'Restaurar Contraseña' (light brown). At the bottom, there is a light gray bar with links: '¿Quiénes somos?', 'Donaciones', 'Términos de Uso', 'Créditos', 'Mapa del Sitio', 'Contacto', and 'Visitas: 14'. Below this bar is a row of five logos: a stylized orange 'B', a blue 'HE', a circular seal, a black 'g', and a blue 'e'.

*Figura 6. Ingreso de usuario*

*Fuente: (Autor 2015)*

## **Vista principal del panel de administración**

En la siguiente figura se muestra el panel de administración de portal.

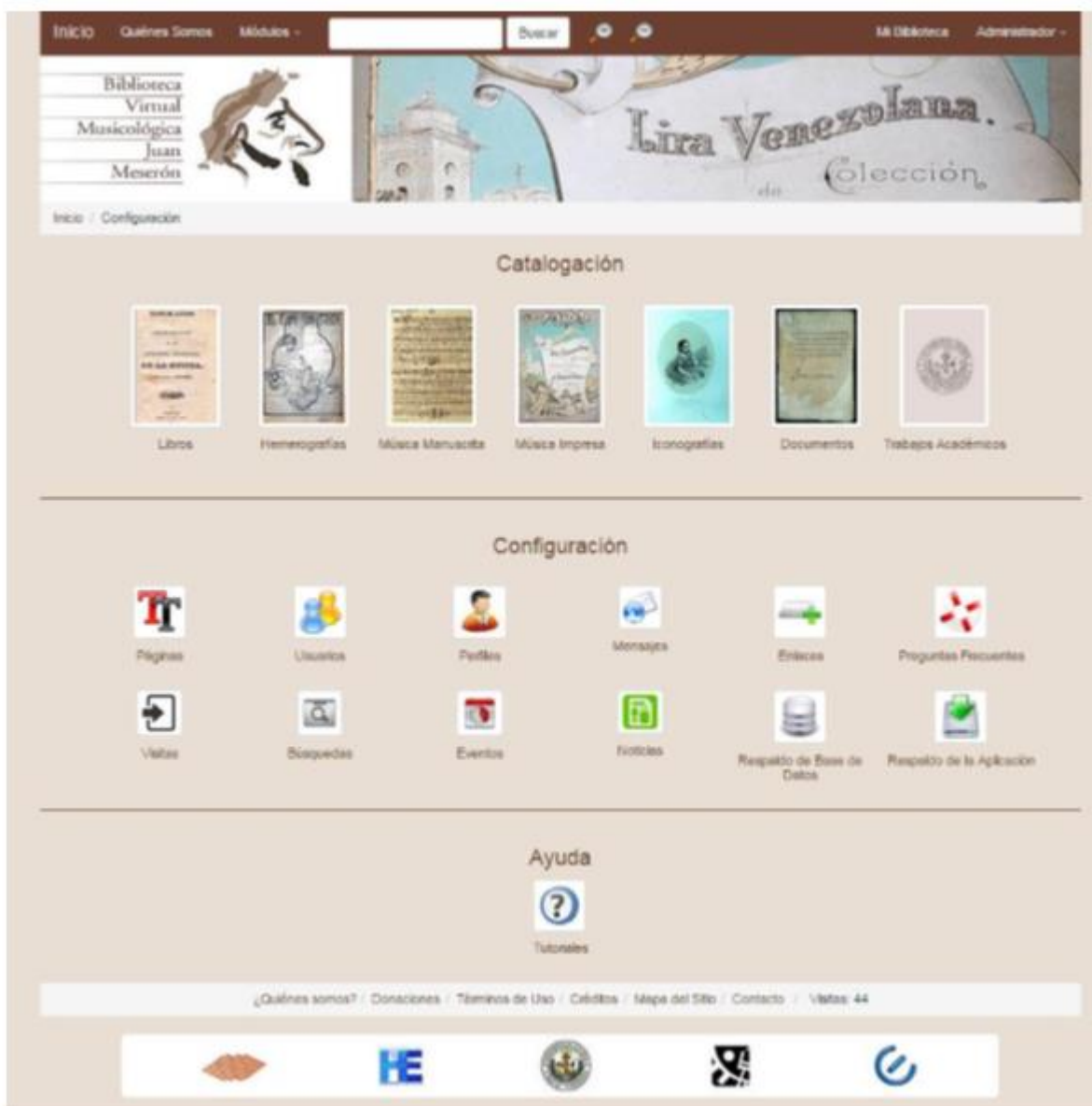


Figura 7. Página principal del módulo de administración.

Fuente: (Autor 2015)

- **Página principal:** aquí se encuentra los accesos a las 18 secciones que ofrece el módulo administrador.
- **Cerrar sesión:** el usuario administrador puede cerrar sesión en la pestaña superior derecha, haciendo clic en ella y seleccionando la opción “Salir”.
- **Editar perfil de usuario:** el usuario administrador puede cerrar sesión en la pestaña superior derecha, haciendo clic en ella y seleccionando la opción “Perfil”.
- **Módulos:** se muestran los accesos a los diferentes módulos presentes en la biblioteca.

- **Soporte de ayuda:** se muestra algunos tutoriales con el fin de asesorar en algunas acciones al usuario administrador y catalogador.
- **Páginas:** le permite al usuario administrador ver, crear, modificar y eliminar las páginas como como donaciones, términos de uso, créditos, mapa del sitio y contacto.
- **Usuarios:** le permite al usuario administrador ver, crear, modificar y eliminar usuarios.
- **Perfiles:** le permite al usuario administrador ver, crear, modificar y eliminar perfiles de usuario.
- **Mensajes:** es una bandeja de entrada de todas las dudas, sugerencias y/o aportes a la página.
- **Enlaces:** le permite al usuario administrador ver, crear, modificar y eliminar vínculos a información importante.
- **Preguntas Frecuentes:** le permitirá al administrador publicar en la Biblioteca Virtual las dudas más comunes entre los visitantes con sus respectivas respuestas. Aún se encuentra en construcción.
- **Visitas:** registra información como dirección IP, fecha y hora, el usuario, el controlador y la acción de cada conexión a la biblioteca.
- **Búsquedas:** registras las búsquedas realizadas en la biblioteca.
- **Eventos:** permite al usuario administrador crear eventos a través de un calendario.
- **Noticias:** sección en donde el usuario administrador gestiona todas las noticias referentes que considere importante.
- **Respaldo de Base de Datos:** permite al usuario administrador realizar un respaldo de la base de datos.
- **Respaldo de la Aplicación:** permite al administrador realizar un respaldo de los archivos de la aplicación y las obras cargadas en el mismo.

### **Vista principal del módulo como usuario Catalogador de Trabajos Académicos**

En la siguiente figura se puede observar la vista principal del Módulo de Trabajos Académicos, desde la vista de usuario catalogador.



*Figura 8. Página principal Módulo de Trabajos Académicos*

*Fuente: (Autor, 2015)*

Al hacer clic en Agregar Trabajo Académico, se mostrará la interfaz para añadir un nuevo Trabajo Académico a la biblioteca usando el formato MARC 21. Así como se muestra en la siguiente figura.





*Figura 10. Detalle Trabajo Académico*

*Fuente: (Autor, 2015)*

Entre las opciones desplegadas en el detalle del Trabajo Académico podemos observar

- **Agregar Trabajo Académico (solo catalogador y administrador):** Un acceso rápido para agregar otro Trabajo Académico.
- **Modificar Trabajo Académico (solo catalogador y administrador):** Un acceso rápido para modificar el Trabajo Académico que se está visualizando en caso de que exista algún error, o por alguna razón el catalogador quiera modificar alguno de los campos.
- **Agregar a Mi Biblioteca (solo usuarios registrados):** Permite a los usuarios agregar el Trabajo Académico que se está visualizando en la biblioteca personal para su posterior acceso.
- **Ver Formato MARC 21:** Despliega toda la información relacionada a la obra en sus respectivos campos del formato MARC 21.
- **Descargue Trabajo Académico (solo usuarios registrados):** Permite a los usuarios descargar el Trabajo Académico si el trabajo en cuestión tiene los derechos de descarga.

## 4.6. Aplicación de la metodología Scrum

En el Capítulo III se describió la metodología a utilizar para la implementación de la aplicación, la cual consiste en una metodología iterativa adaptada al modelo de desarrollo ágil, donde un *sprint* refleja el proceso de creación de un módulo de la aplicación.

En la Tabla 2 se mencionan los *sprints* llevados a cabo para el desarrollo del Módulo de Trabajos Académicos de la Biblioteca Virtual Musicológica “Juan Meserón”.

Sprint	Actividades	Fecha
1	1. Investigar, recopilar y analizar todos los documentos fuentes de información y referencias entregados por la Escuela de Arte. 2. Instalación y configuración del ambiente de desarrollo. 3. Entrenamiento en las herramientas de desarrollo.	20/11/2014 – 15/12/2014
2	1. Diseñar las entidades y las relaciones para el modelo de la Base de datos. 2. Crear una base de datos acorde al modelo previo.	15/12/2014 – 01/01/2015
3	1. Programar las funcionalidades del CRUD (crear, mostrar, actualizar y eliminar) para los Trabajos Académicos. 2. Ajustes en el diseño para su correcta visualización.	01/01/2015 – 20/01/2015
4	1. Desarrollar buscador simple para los Trabajos Académicos.	20/01/2015 – 08/02/2015
5	1. Desarrollar un buscador avanzado para los Trabajos Académicos. 2. Ajustes en el diseño inicial.	09/01/2015 – 20/02/2015
6	1. Pruebas, depuración, optimización y ajustes necesarios. 2. Elaboración del documento del Trabajo Especial de Grado.	21/02/2015 – XX/XX/2015

*Tabla 2 Lista de objetivos por sprint.*

*Fuente: (Autor, 2015)*

## 4.7. Lista de tareas de las iteraciones

### Sprint 1



Las principales actividades realizadas fueron: Instalación, configuración de las rutas, configuración del proyecto en GitHub usando la herramienta Git, configurar la bases de datos, configurar el ambiente WAMP, más adelante cambiado a LAMP, reforzar los conocimientos del patrón de diseño MVC (Modelo Vista Controlador), entender las convenciones en CakePHP basándome en la documentación que se encuentra en la página principal de CakePHP y hacer aplicaciones de prueba para probar las funcionalidades básicas.

### ***Sprint 2***

La principal actividad realizada fue modificar el diseño lógico y físico de la base de datos, añadiéndole los nuevos campos para la catalogación de los Trabajos Académicos. Adicionalmente se comenzó la redacción del documento de este TEG.

### ***Sprint 3***

Se inició formalmente el desarrollo del código referente al Módulo de Trabajos Académicos, se adaptaron los diseños de las interfaces para que fuesen acorde a lo que el módulo está representando. Ajustes como cambio de color, de estructura y de información. Adicionalmente se implementaron las funcionalidades de agregar, eliminar, listar y actualizar los Trabajos Académicos siguiendo el formato MARC 21.

Se dio inicio al proceso de validación de los campos, subcampos y demás elementos a catalogar en el formato MARC 21 asociados a este tipo de trabajo, así como también el diseño de la vista de catalogación. Por otro lado se inició la redacción del documento.

### ***Sprint 4***

La principal actividad en esta parte del desarrollo se enfocó en la programación del buscador básico que permitiera localizar los Trabajos Académicos el cual cumple un papel fundamental en este módulo ya que gracias a él, el usuario puede visualizar la información más relevante de cada trabajo de forma más sencilla y cómoda, es decir visualizar un trabajo por título, por autor, entre otros.

Cumple un papel fundamental en este módulo ya que gracias a él, el usuario puede visualizar la información más relevante de cada trabajo de forma más sencilla y cómoda, es decir visualizar un trabajo por título, por autor, entre otros.

En el momento de desarrollar el buscador surgieron ajustes en los campos a involucrar en el formulario de catalogación para hacer posible el buscador solicitado, es por ello que se agregaron campos en el formulario de catalogación para Trabajos Académicos. Por último se continuó con la redacción del documento.

### ***Sprint 5***

Se desarrolló un buscador avanzado que nos permitiera de manera fácil y rápida acceder los trabajos académicos que se estuvieran buscando tomando como criterios de búsqueda el título, materia, autor o tutor. Para garantizar una búsqueda más especializada con respecto a la que proveía el buscador simple.

Para el buscador avanzado se definieron los principales criterios de búsqueda por el cual se buscarían los trabajos. Se tomaron título, autor, materia, palabras clave, lugar, editor o fecha para realizar una búsqueda más detallada.

### ***Sprint 6***

Una vez desarrollado el buscador avanzado y cumplido con todos los requerimientos iniciales se procedió a realizar una reunión con la profesora Maríantonía donde se notaron algunos detalles que faltaban por corregir tanto a resolver problemas menores, como quitar la opción de descarga de documentos para los usuarios no registrados. Se trabajó en detalles en el filtrado de documentos en la vista principal del módulo y continuó hasta su finalización. Finalmente se continuó el desarrollo del documento.

## **4.8. Pruebas**

Durante el desarrollo del módulo se fueron realizando varios tipos de prueba que fueron necesarias para dar cumplimiento con las solicitudes previas del cliente. Una vez terminado el desarrollo del módulo se acordó una reunión con la tutora por parte de la Escuela de Artes, en la cual se hicieron algunas observaciones en cuanto a la permisología y algunos detalles de interfaz y contenido. Estas observaciones fueron posteriormente incorporadas.

Se hizo uso de un cuestionario como prueba de aceptación adicional. En el Anexo A, se muestra el modelo del cuestionario utilizado. En esta prueba fueron incorporados, algunos desarrolladores de software así como también personas ajenas al área. Se evaluaron tanto los aspectos de funcionabilidad como los de usabilidad en el diseño de las interfaces.

Gracias a las respuestas obtenidas, fue posible evidenciar el nivel de aceptación de los usuarios que fue considerablemente positivo tanto para los aspectos de funcionalidad como de diseño.

#### **4.9. Análisis de Resultados**

Los cuestionarios fueron realizados por un total de 10 personas de perfiles variados, edades comprendidas entre 20 y 56 años y desde personas que poseen un nivel alto de conocimiento de computación hasta personas que generalmente no suelen usar la computadora.

Referente a las funcionalidades pudimos evidenciar un nivel de aprobación del 100%. Los usuarios rápidamente pudieron realizar las diferentes actividades que se propusieron para las pruebas. Entre las pruebas realizadas se abarcaron desde las funcionalidades básicas de un usuario no registrado, hasta la catalogación desde el punto de vista de un usuario catalogador. En el diseño el nivel de aprobación fue del 80%, en cuestión de elección de colores, y diseño de las interfaces.

Adicionalmente se solicitó a los usuarios su opinión referente a los cambios que harían o las funcionalidades que sintieron que faltaron o se podrían agregar para versiones futuras de la Biblioteca. Fueron recabadas esas opiniones y fueron añadidas en la sección de Recomendaciones y Trabajos Futuros.

Como conclusión en base a las pruebas realizadas se arrojaron que la aplicación cumple con los requerimientos planteados por el cliente, dejando abierta la posibilidad de mejoras a futuro por parte de otros tesisistas para los futuros módulos de la biblioteca.

## Conclusiones

El proyecto de la Biblioteca Virtual Musicológica “Juan Meserón” es un proyecto conjunto que están realizando la Escuela de Artes y la Escuela de Computación de la Universidad Central de Venezuela. Cuyo objetivo principal se encuentra en la difusión de materiales bibliográficos y no bibliográficos relacionados con la música en Venezuela.

A lo largo de los últimos años se ha evidenciado una tendencia en el mundo, cada vez hay más personas utilizando el Internet diariamente y usando las tecnologías de información como plataforma de desarrollo para recibir y compartir información de cualquier tipo de manera rápida y sencilla. Las bibliotecas no son la excepción, utilizando recursos como el Internet pueden llegar a un público mucho más amplio que de otra forma probablemente nunca hubiese llegado. Simplemente con una computadora o dispositivos móviles como tabletas y teléfonos inteligentes con conexión a Internet puedes acceder desde cualquier localidad a la información que se está buscando.

En este Trabajo Especial de Grado se desarrolló el módulo de Trabajos Académicos, el cual permite la preservación de trabajos que por su contenido, son de gran valor académico. La forma designada para la catalogación fue MARC 21. También se estableció comunicación con los desarrolladores de los otros módulos para garantizar, de manera práctica la fácil integración de todos los módulos del portal.

Para desarrollar un sistema de esta complejidad, fue necesario adoptar una metodología Scrum que me permitió mantener un proceso de desarrollo eficaz, dinámico y adaptativo.

Gracias a las herramientas, metodología y tecnologías utilizadas; además de contar con la colaboración de un equipo especializado, se lograron los objetivos propuestos.

## Recomendaciones y Trabajos Futuros

En este momento se están desarrollando los módulos “Documentos de la BVMJM” y “Búsqueda Avanzada y el Incipit Musical”. Como recomendación para los tesisistas que están trabajando en dichos módulos están las siguientes.

- Apegarse a los estándares establecidos en las diferentes reuniones y aplicado por los otros tesisistas para así asegurar una fácil integración a pesar de no contar con la presencia física de las personas que trabajaron en los módulos pasados.
- Trabajar usando como base el código actual para garantizar la consistencia tanto de interfaces como de resolución de problemas.
- Para la fecha en la que se publica este Trabajo Especial de Grado, la última versión del código se encuentra en un repositorio público en GitHub, montado inicialmente por el Licenciado Rubén Camargo. Trabajar siempre con esa base y mantenerla actualizada para facilitar el trabajo actual y futuro así sean hechos por otros tesisistas.
- Mantener constante comunicación tanto con la profesora Maríantonía Palacios como con los encargados de La Escuela de Artes como el profesor, Giovani Mendoza, Lars Goldschlager para que el proyecto avance de manera más eficaz y necesite menos correcciones posteriores.

De igual forma como Trabajos Futuros puedo recomendar lo siguiente:

- Retomar los trabajos de internacionalización. Apuntando al éxito de ésta página sería interesante agregar un Módulo de Idiomas que básicamente nos permita seleccionar entre diferentes idiomas, se podría empezar con inglés y español, para que usuarios de diferentes partes del mundo puedan navegar dentro de la Biblioteca Virtual Musicológica Juan Meserón en su idioma nativo.
- Integración con redes sociales: El impacto que han tenido las redes sociales alrededor del mundo es indiscutible. Hoy miles de empresas las utilizan como plataforma de difusión de su contenido. En el caso de la BVMJM se pueden difundir las obras expuestas en la biblioteca añadiendo un botón de compartir para todos los módulos de la biblioteca.

## Referencias Bibliográficas y Digitales

- Agile Manifesto. (2001). Recuperado el 20 de diciembre de 2013, de <http://agilemanifesto.org/iso/es/>
- Alegsa. (2014). Recuperado el 23 de abril de 2014, de <http://www.alegsa.com.ar/Dic/modulo.php>
- Azocar, J. (2010). La biblioteca virtual como medio de búsqueda de la información y el conocimiento para la mejora del hecho educativo. *CONHISREM*, 6(3), 103-104. Recuperado el 5 de abril de 2014, de <http://conhisremi.iuttol.edu.ve/pdf/ARTI000108.pdf>
- Biblioteca Alonso Gamero. (s.f.). Recuperado el 29 de enero de 2014, de <http://biblioteca.ciens.ucv.ve/tesismat.htm>
- Biblioteca del Congreso de EEUU. (s.f.). Recuperado el 29 de enero de 2014, de <http://www.loc.gov/index.html>
- Biblioteca del Congreso de EEUU. (s.f.). Recuperado el 29 de enero de 2014, de <http://www.loc.gov/marc/umbspa/um01a06.html#part2>
- Biblioteca Nacional de España. (s.f.). Recuperado el 29 de enero de 2014, de <http://catalogo.bne.es/uhtbin/webcat>
- Biblioteca Virtual de la Universidad Católica Andrés Bello. (2014). Recuperado el 23 de abril de 2014, de <http://www.ucab.edu.ve/inicio.957.html>
- CakePHP. (2014). Recuperado el 29 de enero de 2014, de <http://book.cakephp.org/2.0/es/cakephp-overview/what-is-cakephp-why-use-it.html>
- CakePHP. (2014). Recuperado el 29 de enero de 2014, de <http://book.cakephp.org/2.0/es/cakephp-overview/understanding-model-view-controller.html>
- CakePHP. (2014). Recuperado el 29 de enero de 2014, de <http://cakephp.org/>
- Camargo, R. (2011). Análisis y Diseño del Módulo “Hemerografía” del Proyecto Biblioteca Virtual Musicológica “Juan Meserón”, Escuela de Artes de la Universidad Central de Venezuela. Trabajo de Seminario. Universidad Central de Venezuela Caracas – Venezuela.

- GIT. (2014). *Acerca del control de versiones*. Recuperado el 28 de enero de 2014, de <http://git-scm.com/book/es/Empezando-Acerca-del-control-de-versiones>
- GIT. (2014). *Fundamentos de Git*, Recuperado el 29 de enero de 2014. de <http://git-scm.com/book/es/Empezando-Fundamentos-de-Git>
- Guzmán, H y Quintana, N. (2011). Análisis y Diseño del Módulo “Libros” del Proyecto Biblioteca Virtual Musicológica “Juan Meserón”, Escuela de Artes de la Universidad Central de Venezuela. Trabajo de Seminario. Universidad Central de Venezuela Caracas – Venezuela.
- Hemeroteca Digital UCAB. (s.f.). Recuperado el 29 de enero de 2014, de [http://200.2.12.132/SVI/hemeroteca/index.php?option=com\\_content&task=view&id=275&Itemid=399](http://200.2.12.132/SVI/hemeroteca/index.php?option=com_content&task=view&id=275&Itemid=399)
- Investic. (s.f.). Recuperado el 23 de abril de 2014. *¿Qué es un CMS o Gestor de Contenidos?*, de <http://www.investic.net/node/90>
- James, M. (2014). Scrum Reference Card. Recuperamos el 15 de enero de 2014, de <http://scrumreferencecard.com/reference-card-de-scrum/>
- JQuery. (2014). Recuperado el 29 de enero de 2014, de <http://jquery.com/>
- Mendoza, G. Procesamiento de Recursos Musicales en Bibliotecas y Centros de Documentación 12 de Mayo de 2014 <dofasolre@gmail.com>
- MySQL Workbench. (2013). *MySQL Workbench*. Recuperado el 1 de octubre de 2013, de [www.mysql.com/products/workbench](http://www.mysql.com/products/workbench)
- MySQL. (2014). *What is MySQL* Recuperado el 27 de enero de 2014, de <http://dev.mysql.com/doc/refman/4.1/en/what-is-mysql.html>
- PC Mag. (s.f.). Recuperado el 28 de enero de 2014, de <http://www.pcmag.com/encyclopedia/term/44707/ide>
- PHP. (2014). Recuperado el 29 de enero de 2014, de <http://www.php.net/>
- Programación. (2013). *Conceptos básicos de ORM*. Recuperado el 5 de junio de 2014, de [http://www.programacion.com/articulo/conceptos\\_basicos\\_de\\_orm\\_object\\_relational\\_mapping\\_349](http://www.programacion.com/articulo/conceptos_basicos_de_orm_object_relational_mapping_349)
- Sabino, Carlos A. (1996). *El Proceso de Investigación*. Editorial Lumen – Humanitas. Buenos Aires.

- Scribd. (2012). *Metodología Ágil vs Metodología Tradicionales*. Recuperado 15 de enero de 2014, de <http://es.scribd.com/doc/91676941/Metodologias-agiles-vs-tradicionales>
- Shore, J.y Warden S. (2008). *The Art of Agile Development*, (1ª Ed), California: OReily. p.9.
- Silberschatz, A. (2009). *Database System Concepts*, (6ª Ed.), New York: McGraw-Hill. p. 1.
- Sublime Text. (2014). Recuperado el 28 de enero de 2014, de <http://www.sublimetext.com/>
- Táchira. (s.f.). *Normas Para El Trabajo De Ascenso Del Personal Académico De La Universidad Nacional Experimental Del Táchira* . Recuperado el 29 de enero de 2014, de [http://secretaria.unet.edu.ve/archivos/D-14\\_Normas\\_Trabajo\\_Ascenso.pdf](http://secretaria.unet.edu.ve/archivos/D-14_Normas_Trabajo_Ascenso.pdf)
- Universidad de Alicante. (s.f.). *¿Qué es un buscador?* Recuperado el 30 de enero de 2014,de [http://www.alu.ua.es/r/rac6/Buscadores/que\\_es\\_un\\_buscador.html](http://www.alu.ua.es/r/rac6/Buscadores/que_es_un_buscador.html)
- Universidad de Córdoba. (s.f.). *¿Qué es un trabajo académico?*. Recuperado el 29 de enero de 2014, de [http://www.uco.es/webuco/buc/pdfdoc/librode\\_estilo.pdf](http://www.uco.es/webuco/buc/pdfdoc/librode_estilo.pdf)
- Universidad de las Américas Puebla. (s.f.). *Tesis*. Recuperado el 29 de enero de 2014, de <http://www.udlap.mx/intranetweb/centrodeescritura/files/notascompletas/tesis.pdf>



## Anexos

### Anexo A

Datos de la prueba	
Fecha de la prueba	
Realizado por	
Ocupación	
Correo electrónico	
Navegador web utilizado	

Prueba de herramientas para usuarios no registrados		
Prueba	Sí	No (especifique el problema)
¿Funciona el buscador simple?		
¿Funciona el buscador avanzado?		
¿Funciona el registro de usuarios?		
¿Funciona el formulario de contacto?		
¿Se puede acceder correctamente al Módulo de Trabajos Académicos?		
¿Funcionan los filtros de índice de Trabajos Académicos?		
¿Se visualiza correctamente el detalle de los Trabajos Académicos?		

Prueba de herramientas para usuario registrado		
Prueba	Sí	No (especifique el problema)
¿Funciona el enlace de descarga del Módulo de Trabajos Académicos?		
¿Funciona en enlace de “Agregar a Mi Biblioteca”?		

Prueba de herramientas para usuario catalogador		
Prueba	Sí	No (especifique el problema)
¿Se pueden agregar Trabajos Académicos a la biblioteca?		
¿Se pueden modificar los trabajos académicos cargados en la biblioteca?		
¿Se pueden eliminar los Trabajos Académicos cargados en la biblioteca?		

Pruebas de interfaz (general)		
Prueba	No	Sí (especifique el problema)
¿Tuvo algún problema de visualización durante su navegación dentro del Módulo Trabajos Académicos?		
¿Considera que hubo alguna función que no fuese intuitiva a durante su navegación?		

**¿Qué aspectos considera que se pueden mejorar en versiones posteriores del módulo?**